Multipath Extension of the ZigBee Tree Routing in Cluster-Tree Wireless Sensor Networks

Zahia Bidai, Oran Es-Sénia University, Algeria
Moufida Maimour, CNRS—Nancy University, France
Hafid Haffaf, Oran Es-Sénia University, Algeria

ABSTRACT

Wireless Multimedia Sensor Networks (WMSNs) are one of the most challenging applications of WSN. They require large amounts of data to be transmitted with high reporting rates which consume an order of magnitude of resources, such as storage, computation, bandwidth, and energy. On the other hand, the ZigBee standard was originally specified for low data rate, low power consumption, and low cost wireless personal area networks (WPANs), making it suitable to WSN. However, handling high data rate applications, such as video surveillance in WPANs, is a challenge. Simultaneous multipath routing is one solution to increase the available bandwidth in a ZigBee network. In this paper, we propose Z-MHTR (ZigBee Multipath Hierarchical Tree Routing), a node disjoint multipath routing extension of the ZigBee tree routing protocol in cluster-tree WSNs. Extensive simulations were performed and showed that the proposed multipath routing enhances application performance in terms of packet delivery ratio, end to end delay, and network lifetime even under heavy data rates.

Keywords: Cluster-Tree Topology, IEEE 802.15.4/ZigBee, Multipath Hierarchical Tree Routing, NS2 Simulations, Wireless Sensor Networks

INTRODUCTION

ZigBee is a robust wireless communication standard managed by the ZigBee Alliance (ZigBee, 2006) and based on the IEEE 802.15.4 physical and MAC layer standard (IEEE, 2006). It defines a network layer, application framework as well as security services. ZigBee aims at handling low data rate, low cost devices and long-life batteries making it very suitable to wireless sensor networks (WSNs) (Akyildiz, Su, & Cayirci, 2002). It can be embedded in a wide range of products and applications. Nowadays, the availability of low-cost CMOS cameras and microphones, Wireless Multimedia Sensor Networks (WMSNs) (Akyildiz, Melodia, & Chowdhury, 2007) gained more interest and research effort. In a WMSN, the scalar WSN is strengthened by introducing the ability of retrieving richer information content through
image and video/audio sensors (Rahimi et al., 2005). This can significantly enhance a wide range of applications like object detection, surveillance, recognition, localization, and tracking. ZigBee supports three types of network topologies namely: star, peer-to-peer and its special case cluster tree topologies. The routing algorithm specified by the network layer of ZigBee depends on the topology used in the ZigBee sensor network. In the simple start topology, there is no routing, data is assumed to be directly transmitted in one-hop to the destination. In the tree topology, a Tree Routing (TR) protocol (Kim, Kim, Yoo, & Lopez, 2007; Nefzi & Song, 2007) is used. Data is routed along the parent-child links established as a result of join operations. Peer-to-peer topology uses a table driven routing basically similar to the Ad hoc On demand Distance Vector (AODV) routing protocol (Baronti et al., 2007; Perkins & Royer, 2003; Karthikeyan, 2010; Ran, Sun, & Zou, 2006). For these two latter topologies, devices can communicate with each other in a multi-hop manner.

ZigBee technology suffers from its limited bandwidth (250 kbps at 2.4 GHz) and extending it to meet WMSN requirements is a real challenge. In the literature, few works already considered multimedia applications over IEEE 802.15.4. In order to guarantee transmitting such dataflow, the authors in (Garcia-Sanchez, Losilla, & Garcia-Haro, 2008) made use of the Guaranteed Time Slot (GTS) (IEEE, 2006) (part of the ZigBee standard). In Deshpande (2006) the GTS mechanism is not used, streaming metrics such as packet loss and latency are analyzed in 802.15.4 networks using the NS2 simulator (Information Sciences Institute, n.d.). A cross-layer solution is proposed in Garcia-Sanchez, Garcia-Sanchez, and Losilla (2010) where the feasibility of transmitting streaming video flows is evaluated. It uses application-level QoS parameters to tune the MAC and physical layers.

In this paper, we explore the use of multipath routing to handle high data rate applications. We chose the cluster-tree topology (IEEE, 2006) of ZigBee and made extensions of the Tree Routing (TR) protocol to allow the formation of multiple disjoint paths. This choice is motivated by the fact that the tree-based topology is efficient, easy to establish and to maintain.

Multipath routing has been one of the most important current directions in the area of routing. It is known for its benefit to Ad-hoc and WSNs. Multipath routing allows the establishment of multiple paths between source and a destination node. According to the application goal, multipath routing may be proposed in order to increase the reliability of data transmission (i.e., fault tolerance) or to provide load balancing and higher aggregated bandwidth. In fact, numerous research investigations on the performance and benefit of multipath routing technique have shown that the use of such routing can improve throughput, reduce end to end delay, increase reliability, ensure security and also mitigate network congestion (Vlajic & Stevanovic, 2010).

Depending on application requirements, the established paths can be used alternatively where only one path is used at a time or simultaneously by using more than one path at the same time. The former approach is more suitable to ensure failure tolerance while the latter may allow load balancing. A multipath routing protocol can fall in one of the following three classes depending on paths disjointness: link disjoint paths, non-disjoint paths and node disjoint paths. However, ZigBee specification does not define the multipath routing mechanism; this limitation has motivated us to propose a solution to this important routing technique in ZigBee WSN which represents the main contribution of this paper.

This paper is structured as follows. The next section presents the related work. Introduced briefly in this section is IEEE 802.15.4/ ZigBee standard and its tree routing scheme. Our proposed ZigBee multipath routing protocol is then described and simulation results are provided. The last section concludes the paper with some future work.
RELATED WORK

Various multipath routing protocols have been proposed for use in multiple applications in WSNs. A multipath routing protocol can be characterized by (i) suitable number of paths with required properties to be achieved, and (ii) the policies on the usage of the different paths and (iii) the traffic distribution on these paths. Multipath routing can be used to improve the robustness of data delivery (Ganesan, Govindan, & Estrin, 2001), reduce the end-to-end delay (Huang & Fang, 2005), achieve load balancing (Wan & He, 2009) or improve the network security (Lou & Liu, 2007), etc. There is not much research on routing dedicated to the ZigBee network layer. The focus was only on AODV and tree routing defined in the ZigBee standard. The aim of most previous work was to improve and enhance the tree routing for ZigBee cluster-tree and mesh networks. In Kim, Kim, Yoo, and Lopez (2007) a shortcut tree routing is proposed to enhance tree routing in ZigBee network through the use of neighbor tables. In this protocol, a source node selects as the next node to transmit packets, the neighbor with the smallest tree level. The use of neighbors table is also described in Nefzi and Song (2007) to make routing decisions. Thanks to the neighbor’s tree address, a node is able to check if a destination is one of its neighbor’s descendant. If so, data is forwarded to that neighbor thus leading to a shorter path. In (Ha, Park, & Kwon, 2007), an Enhanced Hierarchical Routing Protocol (EHRP) is proposed to improve ZigBee tree routing. EHRP uses network addresses and neighbors information to take shortcuts without incurring extra overhead. Authors of (Tsai, Pan, & Tseng, 2009) proposed a new Self-Learning Routing (SLR) mechanism that is more efficient than EHRP (Ha, Park, & Kwon, 2007). SLR inherits the low overhead of tree routing and the path efficiency of mesh routing since it does not send route discovery packets. It routes packets solely based on network addresses and overhearing. Each node maintains a neighbor and rely tables used to take routing decisions.

To the best of our knowledge, there has been no earlier work that deals with multipath routing in ZigBee WSN on top of TR. Instead, there has been only work that addressed the performance evaluation and analysis of various multipath routing in a ZigBee WSN. In Gowrishankar and Basavaraju (2009), the authors considered performance evaluation of four routing protocols, namely AODV, AODVU, RAODV and AOMDV over IEEE 802.15.4 in WSNs using various metrics like packet delivery ratio, average network delay, network throughput and normalized routing load. In Vlajic and Stevanovje (2010), three multipath routing techniques (multipath AODV, multipath DSR and multipath ZDR) are compared to the ZigBee’s standard AODV single path routing protocol. Simulations are conducted in WSNs of different sizes using the IEEE 802.15.4/ZigBee stack provided by OPNET where statistics on end to end delay, packet delivery ratio and battery consumption are collected. Our contribution consists in the design and implementation of a new protocol Z-MHTR (ZigBee Multipath Hierarchical Tree Routing) to build multiple disjoint paths based on three types of information: (i) the MAC parent-child relationships between IEEE 802.15.4 devices in the cluster-tree topology, (ii) the neighbor links maintained by every node using the neighbors table and (iii) the ZigBee tree path information checking multiple paths disjointness property. Extensive simulations have been carried out to get more insight on the behavior of the proposed multipath routing in the context of the ZigBee technology. In addition to the packet delivery ratio, these experiments allow to assess other important metrics such as the end to end delay and network lifetime.

IEEE 802.15.4 AND ZIGBEE

The ZigBee Standard (Zigbee, 2006) provides the network layer specifications to be implemented on top of the PHY and MAC layers standardized by the IEEE 802.15.4 (IEEE, 2006). The IEEE 802.15.4 and ZigBee network
are tightly coupled to provide the consumer standardization for low-power and low-rate wireless communication devices.

**IEEE 802.15.4 Mac Layer**

The MAC layer defines two types of nodes: Reduced Function Devices (RFDs) and Fully Functional Devices (FFDs). An FFD can operate in three different modes: a personal area network (PAN) coordinator, a coordinator or a device. An FFD can also communicate with RFDs and function as a routing device in network topologies where data transfer among FFDs is allowed (e.g., peer-to-peer communication). An RFD is a device with reduced functionality which can only be an end device. It can only associate and communicate with FFDs and does not participate in routing. FFDs and RFDs can be interconnected to form star or peer-to-peer networks. Two modes of operation are mentioned: beacon-enabled and non-beacon-enabled. In the beacon-enabled mode, all the nodes are time-synchronized via regular beacon transmissions from the PAN-coordinator and the coordinators. In the beacon-enabled mode mode, there is no concept of regular beacon transmissions, and thus no time synchronization. The MAC protocol used in the beacon enabled mode is slotted CSMA/CA, while unslotted CSMA/CA is used in the beacon-enabled mode. Moreover, the standard provides the Guaranteed Time Slots (GTS) allocation method in order to provide real time data transmission. There are three types of data transfer mechanisms between network devices: from RFD to FFD, from FFD to RFD and from FFD to FFD. Mechanisms for each transfer type depend on the existence of beacons transmission in the network.

**Zigbee Network**

The ZigBee network layer defines how the network formation is performed and how a network address is assigned to each participant ZigBee device. ZigBee specification extends the basic star topology of an IEEE 802.15.4 PAN to a cluster-tree or a mesh. In a cluster-tree topology, the root, called ZigBee coordinator (ZC), and all internal nodes called ZigBee routers (ZRs), which form clusters, are FFDs. RFDs can only be leaf nodes called ZigBee End Devices (ZEDs). The ZC is responsible for starting a new network. Parent child relationships are established when a ZR or a ZED joins the network. When a parent (ZC or ZR) accepts a node (ZR or ZED) as its child it assigns a unique 16-bit network address to the joining node in the tree. This tree structure is also at the basis of the distributed algorithm for network address assignment. A cluster-tree ZigBee network is characterized by topological parameters \((L_m, C_m, R_m)\). The ZigBee coordinator ZC defines \(C_m\) as the maximum number of children (ZEDs/ZRs) that a parent (ZC/ZR) can have, \(R_m\) as the maximum number of ZRs and \(L_m\) as the maximum depth in the network. According to ZigBee specification, the ZC is at depth 0 and devices at depth \(L_m\) can only be ZEDs. Let \(C_{skip}(d)\), \(d = 0, 1, ..., (L_m - 1)\), the size of the address sub-block being distributed by each parent at depth \(d\) to its router capable child devices (ZRs), is computed as follows:

\[
C_{skip}(d) = \begin{cases} 
1 + C_m \cdot (L_m - d - 1) & \text{if } R_m = 1 \\
1 + C_m - R_m - C_m \cdot R_m^{L_m-d} & \text{otherwise}
\end{cases}
\]

(1)

If the router addresses are numbered from 1 to \(R_m\) and end device addresses are numbered from \((R_m + 1)\) to \(C_m\), for example, a ZigBee device shall be assigned the network address by his parent at depth \(d\) with network address \(A_p\) as follows:

For the \(k^{th}\) router:

\[
A_k = A_p + C_{skip}(d). (k - 1) + 1 \quad (1 \leq k \leq R_m)
\]

(2)

and for the \(n^{th}\) end device:

\[
A_n = A_p + C_{skip}(d). R_m + n \quad (1 \leq n \leq C_m - R_m)
\]

(3)
Zigbee Tree Routing Mechanism

The above calculated addresses determine logical relationships between devices allowing them to make routing decisions in the ZigBee network. In fact, multi-hop routing is among the functionalities provided by this technology. Two routing protocols are considered.

Table-Driven Routing

Table-driven routing is basically similar to AODV. It provides efficient paths formation by broadcasting route request messages and waiting for replies from the destination or intermediate nodes. However, it consumes much communication bandwidth due to the flooding mechanism. Moreover, it may require a large memory space for saving all the discovered paths in the routing tables.

Tree Routing (TR)

It can directly infer the routing paths from network addresses based on the parent-child relationships. No extra memory and broadcast overhead are required. However, it often provides fragile routing paths since it is prone to the single point of failure problem. If one of parent child links in the routing path is broken, it cannot recover the routing path by itself. Moreover, routing paths could be longer because the data packets follow the hierarchical tree topology to the destination even if the destination is close to.

Since tree routing is simple to implement and is lighter in terms of memory and processing routing, it is more suitable for the ZigBee limited resources devices. When a packet is received by a ZigBee router (ZR) at depth \( d \) with address \( A_c \), it decides on whether the next hop node for the destination address \( A_d \) is up or down the tree by applying the routing rules of Algorithm 1.

ZIGBEE MULTIPATH ROUTING

Our proposal consists in \( Z-MHTR \) (ZigBee Multipath Hierarchical Tree Routing), a multipath routing protocol built on top of the ZigBee TR protocol. On one hand, we take advantage of TR’s simplicity and limited required resources. On the other hand, TR’s fragility drawback cited previously can be avoided since other alternative paths can be used in case of path failure. Indeed, in \( Z-MHTR \), more than one path (if available) are used simultaneously, so the aggregate bandwidth may satisfy the bandwidth requirement of high data rate applications. Moreover, these paths are node disjoint which allows additional resources and higher fault-tolerance. In what follows, we begin by giving some assumptions, definitions and concepts before presenting how \( Z-MHTR \) builds its disjoint paths.

Preliminaries

In this work, we only consider the case of routing toward the sink (a central controller or a gateway) of the WSN. The cluster-tree topology is assumed to be rooted at the sink that plays the role of the coordinator. The \( Z-MHTR \) is applied on a steady state network where all the devices are well associated to their parent devices. To make its forwarding decisions, \( Z-MHTR \) relies on this tree structure mainly on the already parent-child established relationships in addition to neighborhood relationships. In \( Z-MHTR \), each node is assumed to maintain an up-to-date neighbors table that contains all nodes that are physically in its radio range. Each entry of the table contains the following elements:

Neighbor identity.
Relationship type: child, parent or adjacent.

This neighbors table can be built simply during joining process when a node scans its neighborhood to find potential parent. This information is used during the routing task. We further assume that neighboring links are symmetric. That is, if node \( N_a \) is a neighbor of node \( N_b \), then node \( N_b \) is a neighbor of node \( N_a \).

The first path used in \( Z-MHTR \) is simply the ZigBee parent-child path.
Algorithm 1. Tree Routing Algorithm

Inputs
\(d\): this router depth, \(A\): this router address, \(A_p\): this router’s parent address, \(A_d\): destination address.

Output
\(A_x\): next hop node address

\[\begin{align*}
\text{if } A < A_p < A + Cskip(d - 1) \text{ then (the destination is downstream)} \\
& \{\text{the packet is forwarded down to an ancestor of the destination node}\} \\
A_x &\to A_j \\
\text{else (destination is upstream and the packet is forwarded to this parent node)} \\
A_x &\to A_p
\end{align*}\]

Definition 1. The ZigBee parent-child path from a source node \(N_s\) at depth \(ds\) to the destination node \(N_d\) at depth 0, via intermediate nodes \(N_{s-1}, N_{s-2}, \ldots, N_1\) is a path denoted by \(ZP = N_s \to N_{s-1} \to N_{s-2} \to \ldots \to N_1 \to N_d\) where all the links are parent-child links.

This first path is built only based on MAC parent-child links. Each node in the path relieves data to its parent until it arrives at the sink.

When forwarding on a subsequent path, a node needs to know to which neighbor transmit a data packet so node disjointness with the already built paths is insured. To do so, based only on parent-child relationships, we introduce the ZigBee Tree Path information.

Definition 2. For a given node \(N_c\) (different from the coordinator) at depth \(dc\) \([1, L_m]\), the ZigBee Tree Path information of node \(N_c\) noted \(ZTP_c\) is an integer sequence \((C_1, C_2, \ldots, C_{d_c})\) that defines the parent-child path in the tree from \(N_c\) to the sink. Each element \(C_k\) \((1 \leq k \leq d_c)\) is simply the rank (based on the ZigBee addresses order) of the child node located at depth \(k\) in the path from the sink to \(N_c\).

A ZigBee tree path information \(ZTP_c\) of node \(N_c\) can takes one of the following forms:

\[ZTP_c = \begin{cases} 
(C_1, C_2, \ldots, C_{d_c}, 0, \ldots, 0) & \text{if } N_c \text{ is an internal node} \\
(C_1, C_2, \ldots, C_{d_c}) & \text{if } N_c \text{ is a leaf node} 
\end{cases}\]

where the first zero value in the sequence indicates termination of the path.

Lemma 1. Each element \(C_k\) of the \(ZTP_c\) where \((1 \leq k \leq d_c)\) and \((1 \leq d_c \leq L_m)\) can be computed using:

\[C_k = \frac{A_c - k - \sum_{i=1}^{k-1} Cskip(i - 1).C_{i-1}}{Cskip(k - 1)} + 1\]

where \(A_c\) is the network address of \(N_c\) and can be deduced using the recursive form of (2) as follows:

\[A_c = (d_c - 1) + \sum_{i=1}^{d_c-1} Cskip(i - 1).C_{i-1} + Cskip(d_c - 1).C_{d_c - 1} + 1\]

Proof: Let
\[ A_c = A_p + Cskip(d_c - 1).(C_{d_c} - 1) + 1 \]

where \((1 \leq C_{d_c} \leq R_m)\), the address of \(N_c\) node which is the \(C_{d_c}\) child at depth \(d_c\) of a parent node at depth \(d_c - 1\). The network address of this parent node is \(A_p\) noted as \(A_{p}^{d_c-1}\) in (4). In its turn, this parent node is also the \(C_{d_c-1}\) child at depth \(d_c - 1\) of a parent node at depth \(d_c - 2\) with \(A_p\) as its network address noted as \(A_{p}^{d_c-2}\) in (4). Since the parent node with address \(A_{p}^{d_c-1}\) must be a router node therefore this address will be calculated by (2) as:

\[ A_{p}^{d_c-1} = A_{p}^{d_c-2} + Cskip(d_c - 2).(C_{d_c-1} - 1) + 1 \]

\((1 \leq C_{d_c-1} \leq R_m)\)

(4)

hence the \(N_c\)’address becomes equal to:

\[ A_c = A_{p}^{d_c-2} + Cskip(d_c - 2).(C_{d_c-1} - 1) + 1 + Cskip(d_c - 1).(C_{d_c} - 1) + 1 \]

By recursively applying (2), the parent address is calculated until it reached 0 which correspond to the coordinator address. At last, the value of \(A_c\) becomes equal to:

\[ A_c = A_p + Cskip(0).(C_{1} - 1) + 1 + Cskip(1).(C_{2} - 1) + 1 + \ldots + Cskip(d_c - 2).(C_{d_c-1} - 1) + 1 + Cskip(d_c - 1).(C_{d_c} - 1) + 1 \]

(5)

Or still

\[ A_c = \frac{\sum_{i=1}^{d_c-1} Cskip(i-1).(C_i - 1)}{C_{d_c} - 1} \]

(6)

From equation (6) and (2) we can deduce the parent address of the node \(N_c\) at depth \(d_c\) computed recursively.

It is given by:

\[ A_p = (d_c - 1) + \sum_{i=1}^{d_c-1} Cskip(i-1).(C_i - 1) \]

(7)

And from (6) we can compute \(C_k\) such that (1 ≤ \(k\) ≤ \(d_c\)) and (1 ≤ \(d_c\) ≤ \(L_m\)) as follows:

\[ C_k = \frac{A_c - k - \sum_{i=1}^{k-1} Cskip(i-1).(C_i - 1)}{Cskip(k-1)} + 1 \]

Note that ZTP components are merely integers and not addresses thus requiring less storage space. Moreover, the first value \(C_1\) of the ZigBee tree path information of node \(N_c\) represents the tree branch number to which belongs the node \(N_c\), where a branch is a subtree rooted at the child number \(C_1\) of the PAN coordinator.

We can see from definitions 1 and 2 that, for a node \(N_c\) at depth \(d_c\), there is a narrow relation between its ZigBee parent-child path ZTP = \(N_c \rightarrow N_{c-1} \rightarrow N_{c-2} \rightarrow \ldots \rightarrow N_i \rightarrow N_j\) and its ZigBee Tree Path information ZTP = \((C_1, C_2, \ldots, C_{d_c})\), that is, the node \(N_c\) is the \(C_{d_c}\) child of the parent node \(N_{c-1}\) which is the \(C_{d_c-1}\) child of \(N_{c-2}\) and so on up along the tree, node \(N_i\) is the \(C_i\) child of the destination node \(N_j\).

**Definition 3.** Two paths \(P = N_i \rightarrow N_{i1} \rightarrow N_{i2} \rightarrow \ldots \rightarrow N_k \rightarrow N_j\) and \(P' = N_i \rightarrow N'_{i1} \rightarrow N'_{i2} \rightarrow \ldots \rightarrow N'_{i} \rightarrow N'_{j}\) are said to be node disjoint paths if and only if there is no common intermediate nodes between them and source \(N_i\) and destination \(N_j\) are the only common nodes (i.e., \(N_i = N'_{i} \forall i, j \in [1, k]\)).
Let \( ZTP_a = (C_1^a, C_2^a, ..., C_n^a) \) and \( ZTP_b = (C_1^b, C_2^b, ..., C_n^b) \) two ZigBee tree path information related to two ZigBee parent-child paths \( ZP_a = N_a \rightarrow N_{a-1} \rightarrow ... \rightarrow N_1 \rightarrow N_d \) and \( ZP_b = N'_a \rightarrow N'_{a-1} \rightarrow ... \rightarrow N'_1 \rightarrow N_d \) respectively of two nodes \( N_a \) and \( N'_a \) located at depths \( d_a \) and \( d_b \) respectively.

**Lemma 2.** If \( C_1^a \neq C_1^b \), then the first and unique common node between the two ZigBee parent-child paths of \( N_a \) and \( N'_a \) is the node with depth 0 (coordinator).

**Proof:** Suppose that the first and unique common node of \( N_a \) and \( N'_a \) has a depth equal to 1 \( \neq 0 \), this means that the two ZigBee parent-child paths of \( N_a \) and \( N'_a \) share a same node at depth equal to 1, that is \( N_1 = N'_1 \) which are two nodes at depth 1, this means, from definitions 1 and 2, that \( N_1 \), respectively \( N'_1 \) is the \( (C_1^a)^{th} \), respectively \( (C_1^b)^{th} \) child of the parent node at depth 0, we deduce therefore \( C_1^a = C_1^b \).

**Lemma 3.** If the first and unique common node of nodes \( N_a \) and \( N'_a \) is the coordinator (sink) with depth 0, then the two paths \( P = N_s \rightarrow N_a \rightarrow N_{a-1} \rightarrow ... \rightarrow N_1 \rightarrow N_d \) and \( P' = N_s \rightarrow N'_a \rightarrow N'_{a-1} \rightarrow ... \rightarrow N'_1 \rightarrow N_d \) are node disjoint.

**Proof:** Suppose that the two paths \( P \) and \( P' \) are not node disjoint, according to definition 3 and due to length path consideration (number of nodes), this means that both paths have in common at least one node with depth not equal to 0.

The following theorem sets the disjointness criterion between two paths.

**Theorem 1.** Two paths 
\[
P = N_s \rightarrow N_a \rightarrow N_{a-1} \rightarrow ... \rightarrow N_1 \rightarrow N_d \quad \text{and} \quad P' = N_s \rightarrow N'_a \rightarrow N'_{a-1} \rightarrow ... \rightarrow N'_1 \rightarrow N_d
\]
are node disjoint paths if and only if \( C_1^a \neq C_1^b \).

**Proof:** Since the two paths \( P \) and \( P' \) are node disjoint then, according to definiton 3, \( N_1 \neq N'_1 \) hence we can deduct, from definitions 1 and 2, that \( C_1^a \neq C_1^b \). If \( C_1^a \neq C_1^b \) then the first and unique common node of \( N_a \) and \( N'_a \) is the node with depth 0 (lemma 2) and by applying lemma 3, the two paths \( P = N_s \rightarrow N_a \rightarrow N_{a-1} \rightarrow ... \rightarrow N_1 \rightarrow N_d \) and \( P' = N_s \rightarrow N'_a \rightarrow N'_{a-1} \rightarrow ... \rightarrow N'_1 \rightarrow N_d \) are node disjoint.

Although using the condition of theorem 1 is sufficient to examine whether two paths are node disjoint, in some situations simpler tests can be used to avoid additional calculations, leading to more efficient implementation. In particular we have the following lemma:

**Lemma 4.** If the adjacent neighbor node \( N'_b \) is a sibling of the node \( N_s \), then the ZigBee parent-child path \( ZP_s = N_s \rightarrow N_{s-1} \rightarrow N_{s-2} \rightarrow ... \rightarrow N_1 \rightarrow N_d \) and \( P' = N_s \rightarrow N'_b \rightarrow N'_{b-1} \rightarrow ... \rightarrow N'_1 \rightarrow N_d \) the path are not node disjoint, where all links except \( N_s \rightarrow N'_b \) are parent child links.

**Proof:** Since \( N'_b \) is a sibling of \( N_s \), they share the same parent node (i.e., \( N_{s-1} = N'_{b-1} \)) that belongs to both \( ZP_s \) and \( P' \) (a common node), we deduct that \( ZP_s \) and \( P' \) are not node disjoint.

Note that from Lemma 4, the second path \( P' \) is usually only one hop longer than the parent-child path \( ZP_s \).

**Node Disjoint Multipath Routing Algorithm**

For a given source node \( N_s \) at depth \( d_s \), Z-MHTR tries to route data packets to the sink (tree root) on multiple disjoint paths. The number of node-disjoint paths is dictated by the network connectivity. It is limited by the minimum number
of neighbors among the source and sink nodes. Routing decisions are performed on-the-fly thanks to the ZigBee tree topology properties without requiring the traditional discovery phase using RREQs. In what follows and for seek of simplicity, we assume that there is three disjoint paths. The generalization to more paths is straightforward. The first path is the one based simply on the parent-child relationships. The basic idea to build subsequent paths is to choose, as the next-hop node, an adjacent neighbor to the source node \( N_s \) and by recursively using the adjacent neighboring or the parent-child links we move in the network until we find an adjacent neighbor node \( N_a \). This latter, must share with \( N_s \) one and only one common ancestor node at depth 0. At this step, the parent-child path from \( N_n \) node to the destination node is taken. In this case the node \( N_n \) belongs necessarily to a different branch (in the tree) which is disjoint to that of \( N_s \).

Based on the multipath transmission policy of the application, the source node \( N_s \) decides to emit a given packet on path \( pNb \) where \( pNb = 1, 2, 3 \) (algorithm 2). For each data packet, we added a new field, called \( \text{flag} \) which can take two values 0 or 1. If \( \text{flag} = 1 \) then \( TR \) is applied on this packet; otherwise new forwarding rules are used.

For data packets to be sent on the first path \( (pNb = 1) \), the source simply applies \( TR \) and set \( \text{flag} \) to 1. This means that a \( TR \) routing is applied on the branch of \( N_s \). To guarantee the node disjointness, only one \( TR \) routing can be applied from a node (for instance, the node \( N_s \)) to the destination node in the branch of that node (for instance, the node \( N_a \)). For the other paths \( (pNb = 2 \ or \ pNb = 3) \), the source node \( N_s \) forwards the data packet to its two adjacent nodes \( N_n1 \) and \( N_n2 \) with \( \text{flag} \) set to 1 if

i. the adjacent node does not belong to the same branch of \( N_s \) and

ii. no \( TR \) routing has been applied on the branch of that adjacent node.

Otherwise it is set to 0.

An intermediate node \( N_n \) simply applies \( TR \) if it receives a data packet with \( \text{flag} = 1 \). If \( \text{flag} = 0 \), based on the depth of the first common node between \( N_n \) and the node which has firstly used \( TR \) routing in \( N_s \)’ branch, the node \( N_n \) chooses the next node using the following rules:

The first common node is at depth \((d_s-1)\). In this case, the common node is not considered as the next-hop node, and an adjacent node \( N_n \) is chosen from the neighbors table. The choice of \( N_n \) is based on:

i. Whether \( N_n \) belongs to a parent-child path or not and

ii. Checking whether the branch of \( N_n \) is already used by \( TR \) routing.

The first common node is at a depth less than \((d_s-1)\). In this case, \( TR \) is applied, this will be done by some intermediate nodes until a parent node at depth \( d_s \) is reached.

If the destination node is in the neighbor table of the current node \( N_s \), directly transmit to corresponding node. Otherwise, apply the multipath forwarding decisions below.

**Illustrative Example:** Figure 1 shows a network topology with parameters \( (L_m, C_m, R_m) = (3, 4, 4) \) where all the sensor nodes are evenly distributed on a square-shaped area. The sink is the ZC located at the center. The mission range (Tr) chosen in this topology allows to cover at least three neighbors and at most eight neighbors. The \( Cskip(d) \) values are respectively 21, 5 and 1 at depth \( d \) where \( d = 0, 1, 2 \) (Equation (1)). For a given node, the two numbers between brackets are respectively the address (that results from equation (2)) and the \( Cskip(d) \) value. The other numbers denote only addresses of nodes. In this example, the source node \( s \) is at depth 2 and has address 17 and nodes \( a, c, b, f \) and \( k \) as neighbors. By applying (7), \( ZTP_s = (1, 4, 0) \) (node \( s \) is the fourth child of node \( a \), which is the first child of node \( d \) (PAN)). The ZigBee parent-child path from \( s \) to the destination node \( d \) is \( s \rightarrow a \rightarrow d \). As shown in Figure 1, three node disjoint paths

---

Copyright © 2012, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
are established from node $s$ to node $d$. The first one is the classical path based on parent-child links and is the shortest one $P_1: s \rightarrow a \rightarrow d$. To construct the second path, node $s$ takes node $c$ as its next-hop node. The intermediate node $c$ at depth 2 compares its $ZTP_c = (3, 4, 0)$ with the source one $ZTP_s = (1, 4, 0)$ and finds that $t = 1$ and the $TR$ routing is not yet applied on the branch of $c$. Consequently, $TR$ is applied from node $c$ until the destination node $d$ which gives path $P_2: s \rightarrow c \rightarrow e \rightarrow d$. The establishment of the third node disjoint path is made in the same manner by taking node $f$ as the next-hop. The resulting node disjoint path is $P_3: s \rightarrow f \rightarrow b \rightarrow g \rightarrow h \rightarrow d$.

**SIMULATION RESULTS**

Different simulations have been carried out to evaluate the performances of the proposed multipath tree routing protocol $Z-MHTR$ with respect to the classical single ZigBee tree routing ($TR$). Simulations are conducted using NS2 simulator with the IEEE 802.15.4 implementation provided by Zheng (2004) composed of the physical layer and the MAC sub-layer of the IEEE 802.15.4 standard in addition to a Service Specific Convergence Sub-layer (SSCS). The SSCS provides an interface to allow higher layers to access services offered by lower layers. We started from this IEEE 802.15.4 implementation and extended it to support neighbors table. Afterwards, we built a network layer on top of the existing SSCS sub-layer in which the ZigBee multipath routing has been developed.

Even if $Z-MHTR$ is able to build more than two paths, we limited our simulations to only two paths. It has been shown that using more than two paths does not achieve further throughput gain (Shyang, Dadej, & Jayasuriya, 2004; Yue Teo, Ha, & Khong Tham, 2007). In all the simulation scenarios, we took $(L_m, C_m, R_m) = (7, 4, 4)$. The traffic generated by the source is Poisson distributed with eight different data rates, $1, 2, 5, 10, 20, 50, 67$ and $100$ pps (packet per second). Each data packet is of $80$ bytes. The data transfer model used is the direct data transmission in a non-beacon enabled network. When a device wishes to transfer data, it simply transmits, using unslotted CSMA/CA, its data frame to the coordinator. This latter sends an acknowledgment frame for each successfully received data packet.

Twelve runs are performed with each time a different randomly chosen source. One simulation is performed for each of the eight transmission rates. Table I summarizes the used parameter values.

The results over all the network scenarios are averaged to produce the following metrics:

*Packet delivery fraction*: the ratio of successfully received data packets at the destination.

*Average end-to-end delay*: the average time for all surviving data packets to go from the source to the destination.
Algorithm 2 Z-MHTR forwarding decisions

Inputs:
The algorithm assumes the existence of three paths. The source has 2 adjacent nodes \( N_1 \) and \( N_2 \).

- \( P_{pN} \): the path number on which the current data packet is to be sent.
- \( N_s \): the source node,
- \( N_p \): this node parent,
- \( N_c \): the current node,
- \( N_n \): the adjacent neighbor node.
- \( B_a \): the branch number of node \( N_a \).
- \( NB_a \): the node which has firstly used the TR routing in branch \( B_a \).
- \( ZTP_{B_a} \): the ZigBee tree path information of node \( NB_a \).

Utilities:

- \( \text{Intersection} (ZTP_a, ZTP_b) \) returns the rank of the first different element between the \( ZTP_a \) of node \( N_a \) and the \( ZTP_b \) of node \( N_b \); if the rank is equal to 1 then the two nodes \( N_a \) and \( N_b \) do not belong to the same branch, otherwise they are in the same branch.
- \( \text{Inclusion} (ZTP_a, ZTP_b) \) returns false if \( ZTP_b \) is not included in \( ZTP_a \) which means that the node \( N_b \) do not belong to the parent-child path of \( N_a \), otherwise it returns true.

Output:

- \( N_x \): the next node.

Forwarding decisions at the source \((N_s = N_c)\)

\[ \text{if} \ (pNb = 1) \ \text{then} \ \{\text{forward on the parent-child path}\} \]

- \( \text{flag} \leftarrow 1; N_s \leftarrow N_p; \ {\text{apply tree routing TR}} \)
- \( N_{B_a} \leftarrow N_s; \ {\text{this means that a TR routing is applied on the branch of}} \ N_s \)

\[ \text{end if} \]

\[ \text{if} \ (pNb = 2) \ \text{then} \ \{\text{forward on the second path (via the first adjacent node)}\} \]

- \( t = \text{Intersection} (ZTP_s, ZTP_{n_1}) \)
- \( \text{if} \ ((t = 1) \ \text{and} \ \{N_{B_a} \ \text{do not exist}\}) \ \text{then} \ \{\text{theorem 1: only one common node at depth 0 (the sink)}\} \)
- \( \text{flag} \leftarrow 1; \)
- \( N_{B_a} \leftarrow N_{n_1}; \ {\text{this means that a TR routing is applied on the branch of}} \ N_{n_1} \)

\[ \text{else} \]
- \( \text{flag} \leftarrow 0; \)
- \( \text{end if} \)

- \( N_s \leftarrow N_{n_1}; \)
- \( \text{end if} \)

\[ \text{if} \ (pNb = 3) \ \text{then} \ \{\text{forward on the third path (via the second adjacent node)}\} \]

- \( t = \text{Intersection} (ZTP_s, ZTP_{n_2}) \)
- \( \text{if} \ ((t = 1) \ \text{and} \ \{N_{B_a} \ \text{do not exist}\}) \ \text{then} \ \{\text{theorem 1: only one common node at depth 0 (the sink)}\} \)
- \( \text{flag} \leftarrow 1; \)
- \( N_{B_a} \leftarrow N_{n_2}; \ {\text{this means that a TR routing is applied on the branch of}} \ N_{n_2} \)

\[ \text{else} \]
- \( \text{flag} \leftarrow 0; \)
- \( \text{end if} \)

- \( N_s \leftarrow N_{n_2}; \)
- \( \text{end if} \)

continued on the following page
Algorithm 2. Continued

If the destination node is in the neighbor table of the current node \( N_c \), directly transmit to corresponding node. Otherwise, apply the multipath forwarding decisions below.

**Forwarding decisions at an intermediate node** \( N_c \),

\[
\begin{align*}
\text{if } (\text{flag} = 1) & \quad \text{then} \\
N_s & \leftarrow N_c; \\
\text{else } (\text{flag} = 0) & \quad \text{then} \\
\text{t} & = \text{Intersection} (ZTP_{B_c}, ZTP_c); \\
\text{if } (t = d_c) & \quad \text{then} \{\text{the first common node is at depth } d_c - 1: \text{lemma 4}\} \\
\text{for } (\text{each adjacent node} \ N_n) & \quad \text{do} \\
\text{if } (N_{B_n} \text{ do not exist}) & \quad \text{then} \{\text{theorem 1}\} \\
\text{flag} & \leftarrow 1; \ N_s & \leftarrow N_n; \ N_{B_n} & \leftarrow N_n; \\
\text{else if } (\text{Inclusion} (ZTP_{B_n}, ZTP_n) = \text{false}) & \quad \text{then} \\
N_s & \leftarrow N_n; \\
\text{end if} \\
\text{end for} \\
\text{else if } (t < d_c) & \quad \text{then} \{\text{the first common node is at depth } < d_c - 1\} \\
N_s & \leftarrow N_c; \\
\text{end if} \\
\end{align*}
\]

**Figure 1. Square-shaped topology**
Network lifetime: the total time during which the network is alive. The network is considered as not operational when the sink stops receiving packets.

To interpret the observed values for the above metrics, we were also interested in the number of dropped data packets due to (i) collisions at the sink, (ii) collisions happened at the nodes belonging to the same path (intra-path collisions), and (iii) intra-path and inter-path interferences.

Packet collision is a situation when two nodes transmit simultaneously, and their packets will collide at recipient node which would not be able to successfully receive any of these packets (Pešovic, Mohorko, Benkic, & Cucej, 2009). Collisions may take place either due to hidden node problem HNP or due to contention medium access. Common practice to avoid these collisions is to use the RTS/CTS handshake mechanism, which is used also in the 802.11 variant of the CSMA/CA protocol. The IEEE 802.15.4 standard (IEEE, 2006) originally does not provide any mechanism to prevent such collisions. In our work, we are firstly interested by the hidden node collisions at the coordinator. We have developed a simple and adaptable mechanism to solve this problem. The hidden nodes are detected at the PHY layer. At the MAC level, the coordinator allocates for each hidden node different sub-periods for transmission in circularly manner using round robin technique. Figure 2 shows that, under different traffic load, our proposed solution to solve the hidden node problem, can achieve significant performance improvement in collisions reduction at the coordinator. The very negligible collisions (about 6 collisions at high traffic load) are mainly due to the contention collisions.

Packet Delivery Fraction: Figure 3 shows the packet delivery fraction to the sink as a function of the inter-arrival rates for the single

<table>
<thead>
<tr>
<th>Table 1. Simulation parameters and settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of nodes</td>
</tr>
<tr>
<td>Radio transmission range (m)</td>
</tr>
<tr>
<td>Number of FFID</td>
</tr>
<tr>
<td>Number of simulation runs</td>
</tr>
<tr>
<td>Number of sources</td>
</tr>
<tr>
<td>Number of sinks</td>
</tr>
<tr>
<td>Sink position</td>
</tr>
<tr>
<td>ZigBee parameters ( L_m, C_m, R_n )</td>
</tr>
<tr>
<td>Initial energy of sensor nodes(j)</td>
</tr>
<tr>
<td>Transmission energy (mW)</td>
</tr>
<tr>
<td>Reception energy (mW)</td>
</tr>
<tr>
<td>Idle energy (µW)</td>
</tr>
<tr>
<td>MAC protocol IEEE 802.15.4</td>
</tr>
<tr>
<td>Propagation model</td>
</tr>
<tr>
<td>Queue size</td>
</tr>
<tr>
<td>Data transfer model</td>
</tr>
<tr>
<td>Traffic model</td>
</tr>
<tr>
<td>Packet size (bytes)</td>
</tr>
<tr>
<td>Simulation area (m × m)</td>
</tr>
<tr>
<td>Traffic load (pps)</td>
</tr>
<tr>
<td>Simulation time (sec)</td>
</tr>
</tbody>
</table>
Figure 2. Number of collisions at the sink in the different routing strategies

Figure 3. Packet delivery fraction in the different routing strategies

Figure 4. Number of intra-path and inter-path interference in the different routing strategies
Figure 5. Number of intra-path collisions in the different routing strategies

Figure 6. Average end to end delay of the different routing strategies

Figure 7. Network lifetime in the different routing strategies
and multipath routing. Basically, for the two strategies routing, the packet delivery fraction decreases as the inter-arrival rate decreases. This is because as the inter-arrival rate decreases the traffic load inversely increases, which in turn results in higher probability that the packets might be dropped due to the increasing number of collisions on the channel and interference. The results show that when the traffic load is in the range of 1 to 20pps all protocols have approximately the same behavior, they are able to handle the traffic quite well, delivering more than 99.5% of the packets to the destination. These results can be explained by the fact that in this range, collisions and interference do not exist for the single path or are very negligible for the multipath routing (Figures 2, 4, and 5). However, once the traffic load exceeds 20pps, the multipath routing, especially M-NIP, outperforms the two other cases. The superior performance of M-NIP is due to the fact that the used paths are less correlated in terms of inter-path interference. Also the use of the mechanism that detects and solves the hidden node problem presented at the sink has enormously improved the performances of the multipath routing.

Average end to end delay: The average end to end packet delay, depicted by Figure 6 increases with the data rate since packets incur more queuing delay. Moreover, due to MAC collisions and interference, packets need to be retransmitted and so their delivery time increases. As in the packet delivery fraction, the end to end delay of the three cases is similar when the traffic load is between 1 and 20pps. However, beyond 20pps, the TR protocol has the worst performance. Here, all the traffic is sent on only one path which leads to an excessive intra-path collisions and intra-path interference (Figures 4 and 5). However, in the multipath routing, where the data packets are spread onto two node disjoint paths, the number of intra-path collisions and intra-path interference is reduced leading to less retransmissions and hence improve the end to end delay.

Network lifetime: For the single-path routing, where the entire traffic load would route along only one path, the sink stops to receive traffic from the source node when the first node along the path dies. On the contrary, in multipath routing, where two node disjoint paths are used and the total network load is distributed evenly into those paths, the sink continues to receive data packets until all the first node of every path dies one after the other. This shows that the use of more than one path prolongs the network lifetime as shown in Figure 7 and hence gives better throughput performance. As expected, the lifetime decreases when increasing traffic load. The simulation results in Figure 7 are consistent with those in Figures 4 and 5. When the traffic load is in the range of 1 to 20pps, the lifetime of all routing techniques is approximately the same. However, when the traffic load exceeds 20pps, a higher volume of packets is transmitted in the network, which will bring more collisions and interference causing an increase in energy consumption to handle all the (re)transmissions of packets. Beyond this saturation value, TR suffers from excessive retransmissions, greatly degrading its performance, while the multipath routing, especially the M-NIP, can still enjoy the integrated capacity of multiple paths and achieves longer network lifetime.

CONCLUSION AND FUTURE WORK

In this paper, we have proposed a multipath extension of the ZigBee hierarchical tree routing Z-MHTR to handle high data rate applications. The proposed Z-MHTR protocol exploits judiciously the address assignment scheme of ZigBee in order to construct node disjoint paths. Different performance studies of the proposed multipath routing are presented and compared to the performance of the classical single-path Zigbee tree routing protocol. First of all, some theoretical results have established disjointness properties of the paths in this protocol. Results showed that, generally, under higher data rates, the performances of all protocols degrade, especially for the single path rout-
ing. Thanks to the proposed solution to avoid hidden nodes collisions at the coordinator (sink) level and the use of two node disjoint paths that minimize considerably inter-path interference, multipath with Non-Interfering Paths (M-NIP) provides the best performances in terms of packet delivery fraction, end to end delay and network lifetime with regard to the two other routing cases examined. Our future work will be focused on the generalization of the hidden node solution on all nodes of the network in order to enhance more application performances. Another application direction is to make the design and implementation of a testbed to perform experiments for evaluation of the performances of our multipath routing protocol for ZigBee cluster tree based WSNs.

ACKNOWLEDGMENTS

This work is partially supported by the PHC Tassili project 09MDU784.

REFERENCES


Zahia Bidai is a researcher-teacher at Oran University, Algeria. She obtained her MS degree (Magister) in computer Science from the Oran University in 2000. She is currently preparing her Ph.D (final phase). Her research interests include routing and MAC in wireless multimedia sensor networks (WMSN) and wireless personal area networks (WPAN).

Moufida Maimour is an Assistant Professor at Henri Poincare University in Nancy, France. She received her engineer degree in computer science from the University of Constantine (Algeria) in 1998. She obtained her Ph.D in 2003 from Claude Bernard University in Lyon, France. She mainly worked on reliable multicast and congestion control in the Internet. Currently, she is making research in the area of wireless sensor networks especially energy-efficient routing protocols, congestion control, mobility and video transport.

Hafid Haffaf obtained Doctor degree in computer Science in 2000; is a senior lecturer at the University of Oran Es-Sénia (Algeria). He also heads the “Industrial computer science and Networks” RIIR Laboratory at Computer science department. His researchers concern different domain as Automatic control, information system but, the more important works are about optimization algorithms, especially those based on graph theory and matroid theory as well as their application in Bond graph and monitoring. He has many collaborations projects with European laboratory: Polytech lille in Engineering and Pau (France) in the domain of Wireless sensor Networks (CMEP project).