RESEARCH   **Open Access**

CrossMark

A QoS-aware dynamic bandwidth allocation scheme for multi-hop WiFi-based long distance networks

Iftekhar Hussain[1*†], Zaved Iqubal Ahmed[1†], Dilip Kumar Saikia[2] and Nityananda Sarma[3]

Abstract

In this paper, we propose a QoS-aware dynamic bandwidth allocation scheme to mitigate congestion problem in gateway-based multi-hop WiFi-based long distance networks and thereby enhance QoS guarantees for real-time traffic. It presents a dynamic slot scheduling scheme which efficiently distributes the unused TDMA time slots among the needy nodes. The distribution process is carried out in a hierarchical manner through the use of parent-child relationship of a tree topology. By doing so, end-to-end performance of real-time traffic is enhanced manifold. Furthermore, the proposed protocol provides assured packet forwarding for already admitted flows by allowing transmission of nodes according to the capacity of ancestral links. Simulation results show that the proposed protocol achieves a significant performance improvement in terms of throughput and delay of real-time traffic.

Keywords: Quality of service (QoS); Real-time traffic; TDMA MAC; Slot scheduling; Bandwidth allocation

1 Introduction

In recent times, WiFi-based Long Distance (WiLD) networks have become very popular in bridging the rural urban digital divide. WiLD links are used to extend Internet connectivity to the remote areas and under-served regions by using only a few number of hops from gateway node [1]. The use of WiFi for covering a rural region is triggered by the wide availability of commodity IEEE 802.11 hardware at low cost and its license-free operation in the ISM band.

Voice- and video-based real-time services have become an indispensable part of today's Internet. The prospective real-time applications over WiLD networks such as videoconferencing in rural telemedicine, e-learning, and voice over IP are required to operate while meeting the user expectations. For example, voice quality of most multimedia services involving voice and video transmission deteriorate dramatically if delay increases beyond a certain limit. Similarly, bandwidth-bound applications involving video streaming expect a minimum level of

throughput guarantee. Therefore, a rural wireless communication architecture must provide some minimum level of quality of service assurance for smooth functioning of real-time applications.

Quality of service (QoS) is a set of qualitative and quantitative traffic characteristics which describe a traffic flow in support of a specific application [2]. Throughput, delay, jitter, packet loss, reliability, availability, etc. are some of the generally considered QoS parameters. Challenges in provisioning QoS over multi-hop WiLD networks are a bit different from traditional wired networks. The existing QoS models do not properly fit into multi-hop WiLD environments due to their architectural differences and several operational constraints. Like other wireless links, the WiLD links are also not reliable due to the factors like signal fading and interference. This unreliability of wireless links create a very dynamic environment where link quality is unpredictable. Moreover, the multi-hop nature of WiLD network greatly affects the end-to-end throughput and delay of already admitted traffic due to intra-flow and inter-flow interference created among the hops. Schedule-based protocols like TDMA are proven to be better solution for provisioning guaranteed bandwidth in WiLD networks

*Correspondence: ihussain@nehu.ac.in
†Equal contributor
[1]Department of Information Technology, North-Eastern Hill University, Umshing, Shillong 793022, Meghalaya, India
Full list of author information is available at the end of the article

Hussain *et al. EURASIP Journal on Wireless Communications and Networking* (2015) 2015:160

Page 2 of 18

[3-5]. In a typical TDMA scheme, scheduling of transmissions aim at increasing the overall network performance. In dynamic traffic situations, provisioning of dedicated bandwidth through TDMA scheduling merely solves the QoS issue. A major challenge in QoS provisioning is to schedule access to the medium based on dynamic traffic demands.

In multi-hop WiLD networks, all the links cannot be allowed to transmit at their maximum capacities even if they can do so. Otherwise, congestion is expected to occur around the root node. Therefore, transmission of the nodes towards the bottom of the tree topology should be restricted in order to avoid congestion near the root node. In such situation, end-to-end QoS provisioning becomes more challenging. MAC protocols proposed for WiLD networks in [3,4,6] do not address this issue, rather they focus on maximizing slot reuse among various neighboring links and thereby improving overall network performance. Although the purpose of multi-hop WiLD and sensor networks are different, they resemble in many aspects particularly in their architectures. Taking cognizance of the congestion possibility, research in sensor networks have developed some interesting MAC protocols such as presented in [7-10] to avoid congestion and hence achieve high end-to-end data rate. To overcome the congestion problem, most of the protocols propose a hybrid MAC combining CSMA and TDMA protocols. Normally, CSMA- and TDMA-based protocols are suitable in low and high traffic load situations, respectively. To assign relative transmission opportunity, several metrics such as distance from the sink to the node, queue length of children node, and node's slot usage history are used. None of the above-mentioned protocols specifically address the QoS issues of real-time traffic. QDBA [11] reserves one part of the TDMA frame for real-time voice traffic and the other part is kept for the dynamic part of real-time video traffic. Dynamic rescheduling of unused TDMA slots to the needy nodes across different levels is therefore important for end-to-end QoS provisioning.

In this paper, we have proposed a dynamic QoS-aware bandwidth allocation scheme, called *DQBA*, to provision QoS for real-time applications in multi-hop WiLD networks. The proposed scheme classifies network traffic into two categories: real-time and best-effort. It takes the bandwidth demands of the nodes in parent-child structure and dynamically schedules channel access based on the received demands. The proposed protocol avoids congestion in the network which is otherwise inherent in multi-hop WiLD networks due to funneling effect [9]. The contributions of this paper can be listed as follows: a) a dynamic bandwidth allocation scheme based on traffic demands is proposed which ensures efficient slot reuse, b) it facilities assured traffic forwarding over

multiple hops through the process of static time slot allocation, c) it provides higher priority to real-time traffic over best effort, and d) a minimum of 20% time slots are kept reserved for each node to avoid node starvation problem. The mechanisms of dynamically reusing time slots in the proposed protocol greatly improves the network performance and provide some luxury to QoS-bound traffic. The simulation results show substantial improvement in throughput and delay for real-time traffic.

The rest of the paper is organized into five sections. Section 2 takes a look on the related works and discusses the different existing TDMA-based MAC protocols proposed for WiLD as well as for sensor networks with reference to their QoS support. The assumptions, design, and algorithms of the proposed protocol have been discussed in Section 3. Simulation results of the proposed dynamic bandwidth allocation scheme and its performance comparison with TreeMAC is presented in Section 4. Finally, Section 5 concludes the paper.

2 Related works

In the literature of WiLD networks, MAC protocols like 2P [3], WiLDNet [4], JazzyMAC [6], JaldiMAC [12], and Lit MAC [13,14] have literally changed the face of WiLD networks. Most of these protocols focus on overall network performance enhancement. Unfortunately, QoS issues in multi-hop WiLD networks are hardly addressed. TDMA-based MAC protocols proposed in [3,4] mainly focus on throughput optimization by generating optimal TDMA schedule considering efficient slot reuse. JazzyMAC [6] assigns variable length transmission slots according to the traffic demands of nodes. This protocol is specifically designed to allow neighbors to proceed with parallel independent transmissions without waiting for the marker packet to arrive. It causes increased throughput performance. Unlike the 2P-based MAC protocols, JaldiMAC supports single-hop point-to-multipoint network architecture which relies on loose node synchronization. It allows dynamic traffic patterns with varying symmetry ratios to adapt with the asymmetry of Internet traffic and allocates transmission slot based on demands. However, JaldiMAC cannot be scaled up for multi-hop topology with ease and will have similar problems as JazzyMAC.

Sensor network topology is predominantly tree like; mostly with low bandwidth and short distance links. Unlike traditional wireless networks, wireless sensor networks (WSNs) consist of spatially distributed autonomous sensors which do not need to communicate directly with the nearest high-power base station. Rather, the sensor nodes cooperatively pass their data through the network to a central location, called sink node, by communicating with their local peers. The nodes forward traffic

Hussain *et al. EURASIP Journal on Wireless Communications and Networking* (2015) 2015:160

Page 3 of 18

hop-by-hop as such the major traffic patterns are many-to-one forming a tree [7]. Thus, the sensor network architecture has a phenomenal similarity with multi-hop WiLD networks although traffic characteristics are quite different.

In tree-based networks, nodes closer to the root need to forward more data packets than others. Traditional MAC protocols tend to provide fair access and hence are not suitable in such network architecture. If traditional MAC protocols are used in many-to-one network topology, congestion towards the root node is inevitable. Recognizing this fact in WSN, a number of protocols such as presented in [7,8,11,15-18] are developed. A hybrid approach using schedule-based medium access in traffic-intensive regions and contention-based MAC in low traffic zones is proposed in [8,15,17]. Z-MAC [8] acts like a contention-based protocol under low traffic conditions and a schedule-based protocol under high traffic conditions by using the schedule computed by DRAND (Distributed RAND) [19]. It allocates time slots to every node ensuring that no two nodes among a 2-hop neighborhood are assigned the same time slot. In order to improve utilization in low load situation, Z-MAC allows 'non-owners' of a slot to contend for a slot if it is not being used by its 'owner'. Similarly, Funneling-MAC [15] tried to mitigate the funneling problem by a sink-oriented scheduling protocol which is also a hybrid of TDMA and CSMA protocols. It uses TDMA scheduling in the intensity region and employs CSMA in the rest of the network to provide flexibility. It is localized in operation because TDMA only operates in the intensity region close to the sink and not across the complete sensor network. These two protocols employ fixed slot TDMA and hence do not provide any priority to the nodes considering their requirements. I-MAC [16] assigns different levels of priority to different nodes according to their role in the network. During scheduling of any slot, the owner of the slot gets the first priority. The non-owner nodes can compete to use a slot only when the owner node does not need it. The chance of getting a slot by a non-owner node also depends on its priority level. Queue-MAC [10] is another hybrid protocol which has addressed the issue of burst network traffic by allocating time slots of dynamic size. In this protocol, packets coming from the children nodes carry their load information through a special field called *queue indicator*. The frame comprises a CSMA and a TDMA component. Initially, a node starts its transmission using CSMA protocol. With an increase in load, the active TDMA period is accordingly extended by adding more time slots to increase the bandwidth. Queue-MAC considers only single-hop topology because of which it needs multi-hop extension to fit in WiLD networks.

TreeMAC [7] attempted to solve the congestion problem by using only TDMA-based MAC for the entire WSN.

In this protocol, time slots are allocated to the nodes following a two-dimensional approach. A time cycle is divided into frames and each frame into slots. A parent node determines children node's frame assignment based on their relative bandwidth demand, and each node calculates its own slot assignment according to its hop distance to the sink. Each children node notifies its parent of its bandwidth demand by piggybacking this information in a routing beacon message. Making use of queue length of all the sensor nodes, iQueue-MAC [17] assigns TDMA slots of variable size for packet transmission. iQueue-MAC uses CSMA and TDMA protocols in light and high load situations respectively. It integrates a variable namely *TDMA period* to provide adaptive data transmission based on children node's queue length. Utilization-based scheduling [18] used Spatial-TDMA (STDMA)-based dynamic channel access mechanism to increase throughput in wireless mesh networks. In accordance with the node's slot usage history and packet-queue occupancy, each node is assigned a dynamic weight value which approximates the node's demand for transmission slots in the next frame. The number of times TDMA slots assigned to each node in a single frame is proportional to its weight. To allocate bandwidth dynamically for real-time traffic, a QoS-aware dynamic bandwidth allocation (QDBA) scheme for WiMAX-based networks is proposed in [11]. QDBA scheme divided a TDMA frame into two parts, one is steady part for real-time voice traffic and the other is the dynamic part for real-time video traffic. The base station allocates bandwidth to the subscriber stations based on the QoS requirements of the connections.

The MAC protocols discussed above have mostly addressed the congestion issue by employing a TDMA-based MAC at high traffic load. TDMA slot assignment is further improved by incorporating adaptive or variable TDMA which allocates time slots based on demands. The traffic demand is decided from the role of a node in the network and explicit information such as queue length, slot usage history, etc. However, none of the above protocols consider provisioning QoS for the real-time traffic with dynamic bandwidth allocation in a precise manner. This motivates us in designing a dynamic bandwidth allocation scheme which will ensure end-to-end QoS for real-time traffic.

3 The proposed dynamic bandwidth allocation scheme

In provisioning QoS for real-time traffic, the proposed scheme aims at meeting the QoS demands for upward traffic which usually suffers from congestion close to the gateway node due to funneling effect [15]. The proposed scheme works in two phases. In the first phase, guaranteed TDMA slots are statistically allocated to all the

Hussain *et al. EURASIP Journal on Wireless Communications and Networking* (2015) 2015:160

Page 4 of 18

nodes of the network. It regulates various links' transmission by introducing an operational constraint called *restricted simultaneous operation (R-SynOp)*. In the second phase, transmission opportunities for QoS-bound traffic are enhanced through sharing of the unused bandwidth dynamically among the needy nodes. In this section, we have discussed the assumptions, structure of the TDMA frames, and the design of proposed scheme in details.

3.1 Protocol assumptions

In multi-hop configuration, wireless nodes are equipped with multiple radios co-located on the same tower. We consider a tree topology in our model which can easily be formed either by constructing it physically or by logically converting the mesh/graph topology. Figure 1 depicts a typical tree topology using which various important concepts involved in the proposed scheme is explained. Most of the existing MAC protocols for such networks use a token-based mechanism to establish a loose synchronization in a part of the network without any concern over the entire network synchronization [3,4,6,12]. The proposed scheme assumes tight node synchronization in the network. Nodes at the same level of the tree topology do not interfere with each other as it uses point-to-point directional links. Hence, we do not consider any horizontal interference which exists in networks using omni-directional antennas as mentioned in [7].

In WiLD networks, a node equipped with multiple radios can use separate non-overlapping channels on different links. However, in the line of the schemes proposed in [3,4,6], we consider the use of a single channel in our design. Each node employs a separate radio for all of its point-to-point links. All the co-located radios in a node can perform simultaneous synchronous transmit (SynTx) or simultaneous synchronous receive (SynRx) operation at the same time but mix of both is not achievable. This mode of operation is termed as *synchronous operation (SynOp)* [3]. This combination of hop-by-hop communications and data forwarding to root node creates a choke point on the free flow of traffic particularly in the nodes nearer to the root. The funneling effect [9] of traffic leads to increased transit traffic intensity and delay as events move closer towards the root node, resulting in significant packet collision, congestion, and loss leading to collapse of the network. Funneling effect is explained in Figure 1. For upward traffic, allowing all the children nodes to transmit simultaneously by performing SynOp may introduce congestion in the ancestral links which in turn will degrade the network performance. To overcome such a situation, the sibling nodes should not be allowed to perform SynTx simultaneously in forwarding traffic towards their parent. In the proposed scheme, an enhanced version of SynOp termed as *Restricted-SynOp* (R-SynOp) has been introduced which restricts the simultaneous sibling node transmission in the upward direction, i.e.,
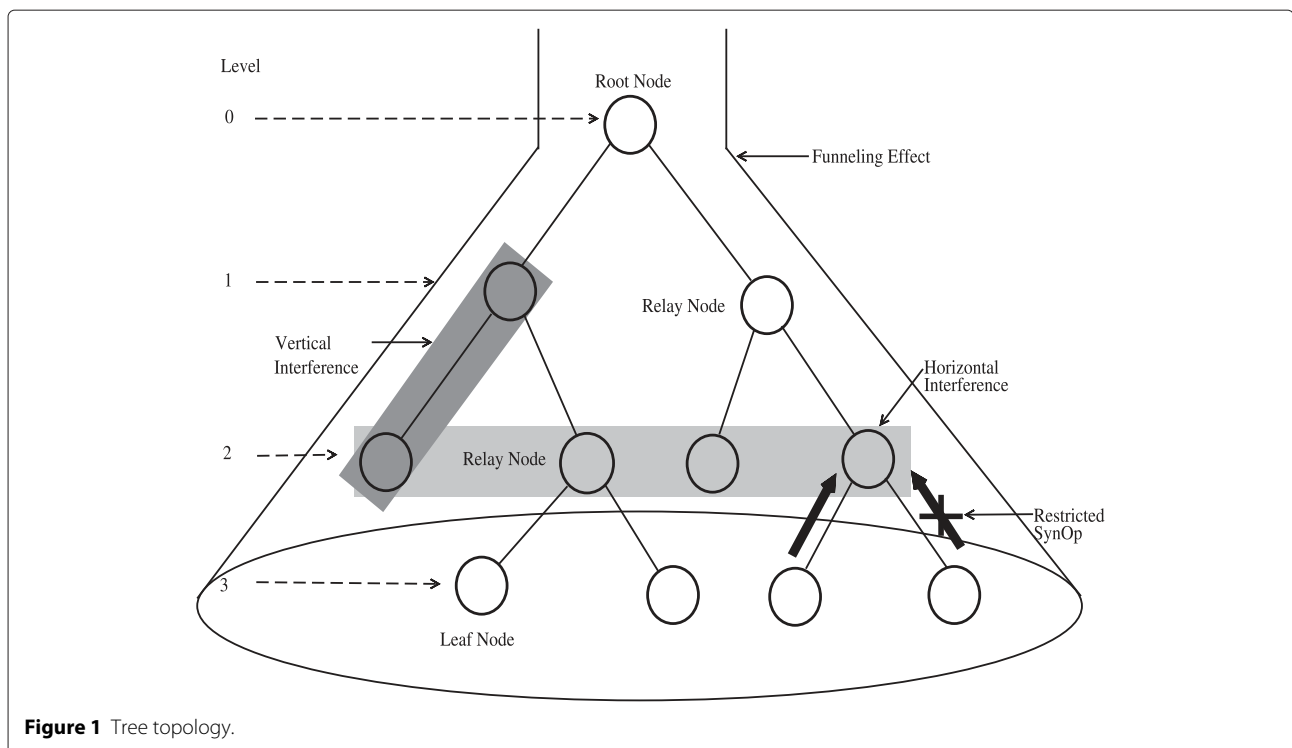

Figure 1 Tree topology.

Hussain *et al. EURASIP Journal on Wireless Communications and Networking* (2015) 2015:160

Page 5 of 18

children to parent node. Thereby, it prevents the parent node from being congested. However, the use of restricted SynOp in the first-level nodes is an exception as the root node is assumed to be connected with very high bandwidth link. The concept of R-SynOp is demonstrated in Figure 1. Use of R-SynOp does not impact the overall network performance and hence allows smooth flow of traffic.

3.2 Frame structure
In this scheme, a TDMA frame is broadly divided into two parts: synchronization interval and service interval. Figure 2 describes the format of a superframe indicating the distinct elements of it.

3.2.1 Synchronization interval
The synchronization interval comprises of control and contention slots which are used to synchronize the nodes in the network. The control slots are used to transmit control information such as TDMA frame and bandwidth grant. The number of control slots required to disseminate control information to all the nodes in a network is equal to the depth of the topology tree. Contention slots are used only by the non-root nodes (i.e., relay and leaf nodes) for the purpose of sending node join requests and bandwidth demands.

3.2.2 Service interval
Service interval is the time elapsed between two consecutive synchronization intervals during which data

transmissions are scheduled. The service interval is equi-partitioned into unit slots which are necessarily even in numbers. Since transmissions of all the nodes of a network can be scheduled within two slots, say T_i and T_j where $i \neq j$, we have merged two consecutive unit slots to form a time cycle, T_{cycle}. Hence, the service interval can be visualized as a collection of time cycles.

3.3 The protocol
Let a multi-hop WiLD network be represented as a tree $T = (V, E)$ where, V is the set of nodes and E is the set of links in the network. Let $n \in V$ be any arbitrary node. Adj[n] and Child[n] represent the number of adjacent links and the number of children nodes of the node n, respectively. The protocol distinguishes three different types of nodes in the network and entrusts different responsibilities to them. The different categories of nodes are: i) root node, ii) relay node, and iii) leaf node.

The root node, R, is a special node which satisfies the condition Adj[R] = Child[R] and acts as the central coordinator of the network. It carries out the task of constructing TDMA superframe, generating control packets, disseminating TDMA frames, multi-hop node synchronization, and static slot allocation process. Relay node receives control packets, processes, and forwards them to their children nodes. In addition, it forwards node join request to its parent and initially carries out the static slot allocation process for its 1-hop children nodes too. These nodes perform dynamic slot assignment on receipt of the
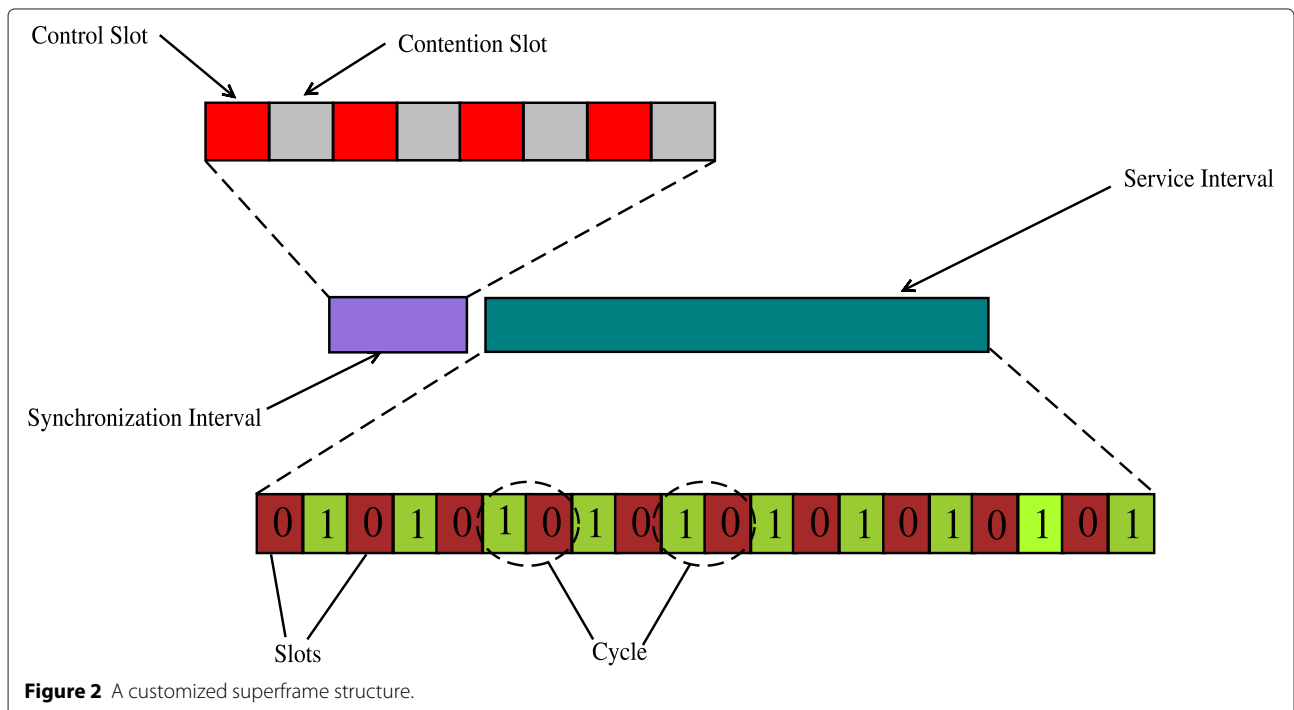

Figure 2 A customized superframe structure.

Hussain *et al. EURASIP Journal on Wireless Communications and Networking* (2015) 2015:160

Page 6 of 18

traffic demands from their children nodes. A leaf node does not have any children and always has exactly one adjacent link. Leaf nodes are the end-points of the network which carry out the task of receiving control packets, generating node join request, and sending bandwidth demands to their parent nodes.

In the proposed scheme, we logically divide a multi-hop WiLD network into 1-hop clusters as shown in Figure 3. All the clusters exhibit a parent-child relationship and have identical behavior except at the cluster containing the first-level nodes. The kth cluster in level l consists of r_k^l number of nodes in such a way that $(r_k^l - 1)$ nodes are 1-hop children of a given parent node. Let us assume that at a given level l in the network, there are m number of clusters. The parent node in the kth cluster of level l is represented as $n_{p_k}^l$ and the ith children in that cluster is represented as $n_{i_k}^l$ such that $0 < i < r_k^l$.

The initial slot allocation and dynamic QoS-aware slot scheduling phases of the proposed scheme are explained below.

3.3.1 Initial static slot allocation

During the network initialization, a static slot allocation process is carried out. The basic task behind this slot allocation process is to equally distribute the time slots of a parent node among its children nodes of a given cluster. If the parent node, $n_{p_k}^l$, of a cluster has a transmission slot T_i in a given time cycle, T_{cycle} where $i \in \{0, 1\}$, the other slot in the T_{cycle}, i.e., T_j, where $j \in \{0, 1\}$ and $T_i \neq T_j$, can be

occupied by any other children node of that cluster. If T_j is allocated to node $n_{i_k}^l$, no other sibling nodes of $n_{i_k}^l$ shall be assigned the same time slot for transmission to their parent node, $n_{p_k}^l$.

The root node which acts as the central coordinator starts the slot allocation process. With the assumption that the root node has greater transmission capabilities in forwarding traffic outside the network, the root node shares the T_{cycles} equally between itself and all of its 1-hop children nodes. Therefore, if the total number of slots in a TDMA superframe is γ, the slot share of each 1-hop children can be given by

$$\gamma_{n_{i_k}^1} = \frac{\gamma}{2}$$

Here, $\gamma_{n_{i_k}^1}$ represents the bandwidth share of ith node belonging to kth cluster of level-1.

Now, as all the 1-hop children of the root node get the slot share equal to their parent, they can further share half of their allocated slots among their children. The slots allocated to a node need is shared equally among its children nodes. Therefore, the bandwidth share received by the ith children in kth cluster of level l denoted as $\gamma_{n_{i_k}^l}$ can be given by

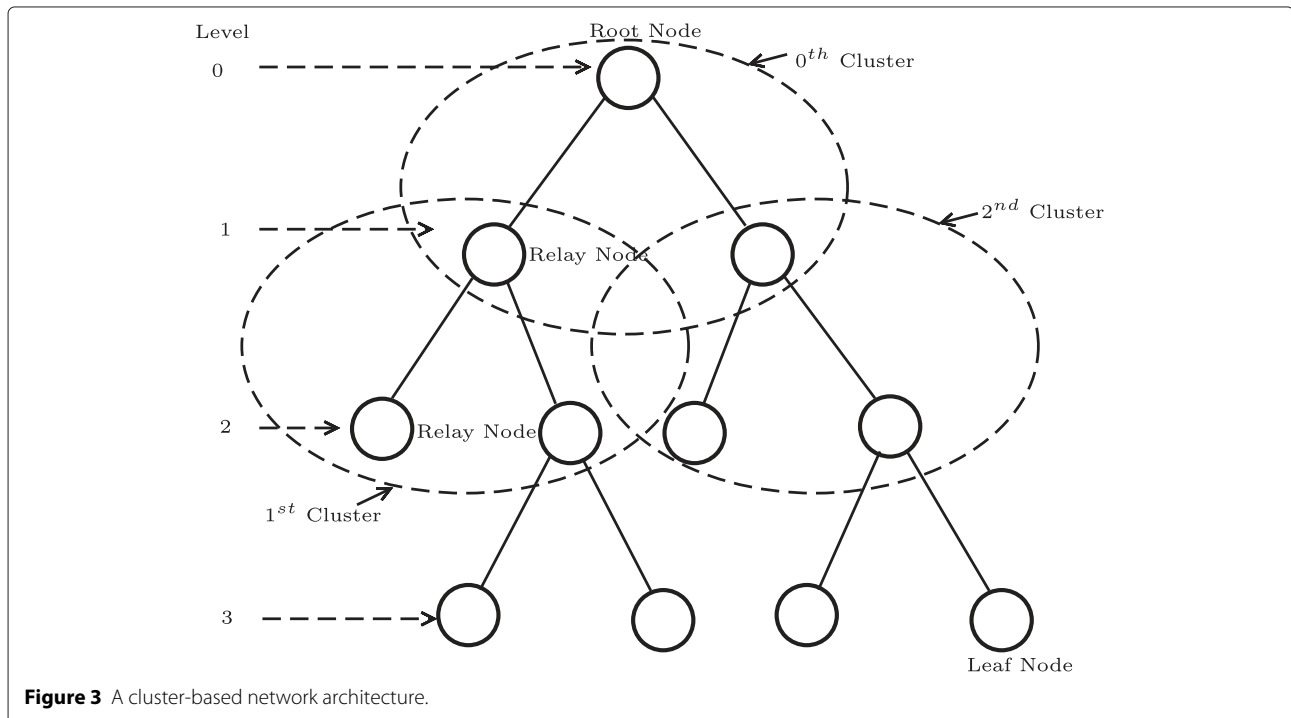$$\gamma_{n_{i_k}^l} = \frac{\gamma_{n_{p_k}^l}}{r_k^l}$$

Figure 3 A cluster-based network architecture.

Hussain *et al. EURASIP Journal on Wireless Communications and Networking* (2015) 2015:160

Page 7 of 18

Algorithm 1 Static slot allocation algorithm

<div align="center">

Input:

T_{fr}: TDMA Frame

n_k: Parent of kth Cluster

$Child[0..(n-1)]$: Children of n_k

</div>

1: **if** (n_k = Root Node) **then**
2:     **for all** T_{cycle} in T_{fr} **do**
3:        Assign T_i to n and T_j to $Child[0..(n-1)]$
    such that $T_i \neq T_j; 0 \leq i,j \leq 1$
4:     **end for**
5: **else**
6:     **for all** T_{cycle} in T_{fr} **do**
7:        **if** T_i is assigned to n_k **then**
8:          Assign T_j to any one of the nodes in
    $Child[0..(n-1)]$ such that $T_i \neq T_j; i \geq 0; j \leq$
    1 in round-robin fashion
9:        **end if**
10:     **end for**
11: **end if**

This static slot allocation process is carried out subsequently by all the nodes which has at least one children node. The initial slot allocation process is illustrated in Algorithm 1. The algorithm first checks whether a node is root or non-root and then it starts allocating slots. The root node allocates every next slot in a frame to itself and the remaining to all its children nodes, thus sharing equal bandwidth between itself and its 1-hop children nodes. In the case of a non-root node, every alternate slot is allocated to one of the children nodes in a round-robin fashion. This algorithm ensures that no two nodes at 1-hop distance get the same slot for transmission which would otherwise have resulted in 1-hop vertical interference.

3.3.2 Dynamic slot scheduling

The proposed protocol introduces a dynamic slot allocation scheme based on bandwidth demands of the children nodes. Bandwidth demands of children nodes are placed to parent node through sending traffic indication map (TIM). After receiving the TIMs from its children, a parent node tries to allocate time slots according to their demands. If a parent node is not able to allocate the required number of slots to its children nodes, it prepares a TIM specifying the requirement and sends to its immediate parent node. This protocol classifies the demands of the children nodes into two broad categories - QoS demand (*Q-demand*) and additional demand (*A-demand*). The Q-demand indicates the total bandwidth demand for the delay and bandwidth sensitive real-time traffic such as VoIP and videoconferencing whereas A-demand indicates the requirement of non-real-time (best-effort) traffic. Both the demands are specified in terms of number of slots. The TIMs are

sent in the last slot allocated to the node for transmission either through special packets or by piggybacking in data packets. In the absence of slots for transmission, the TIMs are sent in contention slot allocated to a node.

On receipt of TIMs from all the children, a parent node starts the dynamic slot rescheduling process and tries to fulfill the bandwidth demands while preparing transmission schedule for its children. The rescheduling process prioritizes Q-demands over A-demands. Therefore, a parent node first schedules the Q-demands of all of its children and then tries to schedule the A-demands. After generating the schedule, the parent node sends it to all of its children nodes. The scheduling process is highly localized where a parent prepares transmission schedules for its 1-hop children and distributes it without burdening or influencing other nodes with the exception in certain situations.

Consider cluster k belonging to level l. Let the slot allocated to parent of the cluster, $n_{p_k}^l$, be $\gamma_{n_{p_k}^l}$ and the demand from ith children of the cluster be $\beta_{n_{i_k}^l}$. Then, the slots allocated to a children node, $n_{i_k}^l$, can be calculated by using Equation (1).

$$
\gamma_{n_{i_k}^l} = \min\left(\beta n_{i_k}^l, \gamma_{n_{p_k}^l} \times \frac{\beta_{n_{i_k}^l}}{\sum_{j=0}^{r_k^l} \beta_{n_{i_k}^l}} \right) \tag{1}
$$

In situations when the Q-demands of all the children of a node is less than the allocated bandwidth, it releases the additional bandwidth to its parent in order to enable the use of those unused slots by other nodes. In such cases, a maximum of 80% of the total available bandwidth are released. The remaining 20% are reserved for future communication by the node. This phenomenon is termed as *bandwidth release*. On the other hand, when the total demand of all its children exceeds the allocated bandwidth, the available bandwidth is shared among its children based on their demands and the additional bandwidth request which could not be served are sent to its parent. The process of requesting additional TDMA slots is similar to that of a children placing traffic demands to its parent. Dynamic slot scheduling process is eventually started by the leaf nodes and continues till 1-hop children of the root in a hierarchical fashion.

Algorithm 2 presents the working of the dynamic slot allocation process. The cluster head of cluster k, n_k is distributing a total number of γ_{n_k} slots of a TDMA frame, T_{fr} among its children. The algorithm first checks in which frame the node has a slot allocated to it and proceeds with allocating the other slot to any one of its children. The bandwidth γ_{n_k} is divided between Q-demand and

Table 1 Simulation parameters

Parameter Name	Value
Traffic types	CBR, VBR
Packet size (CBR)	1,250 bytes (payload)
Packet interval (CBR)	33 ms
Routing protocol	Fixed routing protocol
Simulation area	50×50 km flat-grid area
Radio propagation model	Two-ray ground reflection model
Bandwidth	11 Mbps
Antenna type	Grid parabolic antenna
Antenna gain	24 dBi
Distance per hop	9 km
Number of Nodes (Max.)	8
TDMA slot time	4 ms
Guard time	100 μs
TDMA queue length	100

A-demand traffic in the ratio 80:20. The time slots allocated for Q-demand and A-demand traffic are represented by $\gamma_{n_k}^Q$ and $\gamma_{n_k}^A$, respectively. After receiving the Q-demand (α_i) and A-demand (β_i) from all of its n children nodes, Child[$0..(n-1)$], the parent node computes the cumulative Q-demand, α^c, and A-demand, β^c. Out of the available slots in $\gamma_{n_k}^Q$, it serves the Q-demands of all of its children nodes first. After serving the Q-demands, the A-demands of all the children nodes are served from the 20% bandwidth earmarked for A-demand, $\gamma_{n_k}^A$. Any slots remaining unused after slot allocation for Q-demands are also allocated to A-demands.
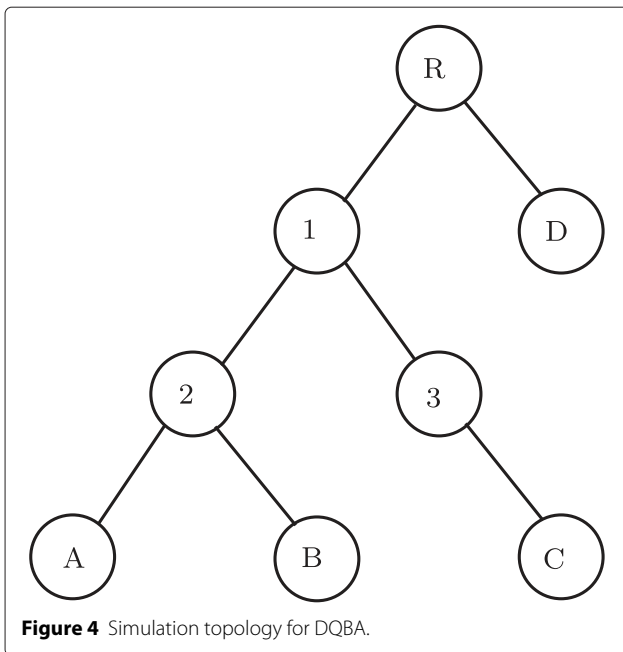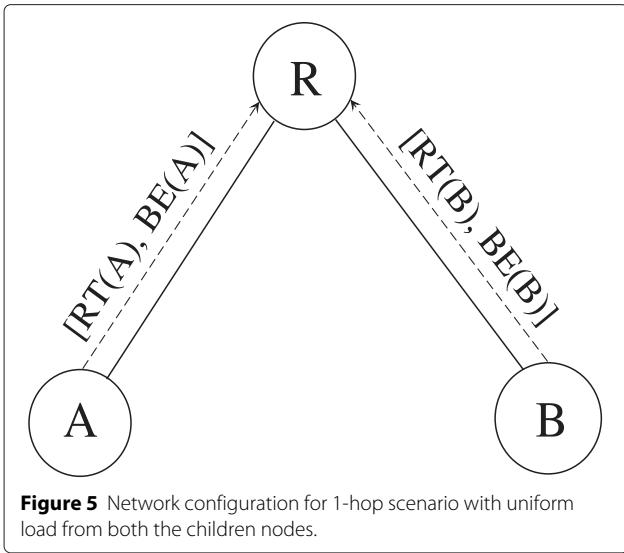

Figure 4 Simulation topology for DQBA.

Algorithm 2 Dynamic slot scheduling algorithm

Input:
T_{fr}: TDMA Frame
γ_{n_k}: Slots assigned to n_k
α_i: Q-demand of ith node
β_i: A-demand of ith node

1: $\gamma_{n_k}^Q \leftarrow 0.8 \times \gamma_{n_k}; \gamma_{n_k}^A \leftarrow 0.2 \times \gamma_{n_k}; n_{child}$: child number
2: $i \leftarrow 0; \alpha^c \leftarrow 0; \beta^c \leftarrow 0; n_c hild \leftarrow 0$
3: **while** $i < n$ **do**
4: 　　$\alpha^c \leftarrow \alpha^c + \alpha_i$
5: 　　$\beta^c \leftarrow \beta^c + \beta_i$
6: 　　**increment** i
7: **end while**
8: **for all** T_{cycle} in T_{fr} **do**
9: 　　**if** T_i is allocated to n_k **then**
10: 　　　　$m \leftarrow (n_c hild ++) \bmod n$
11: 　　　　$r \leftarrow 0$
12: 　　　　**while** $r < n$ **do**
13: 　　　　　　**if** $\left(\alpha_m > 0 \ \& \ \gamma_{n_k}^Q > 0\right)$ **then**
14: 　　　　　　　　Allocate $T_{(i+1)\bmod 2}$ to $Child[m]$
15: 　　　　　　　　**decrement** $\alpha_m, \gamma_{n_k}^Q$
16: 　　　　　　　　**if** $\gamma_{n_k}^Q = 0$ **then**
17: 　　　　　　　　　　RESOURCE_REQUEST($\gamma_{n_k}^Q, \gamma_{n_k}, \alpha^c, \beta^c$)
18: 　　　　　　　　**end if**
19: 　　　　　　　　**break loop**
20: 　　　　　　**end if**
21: 　　　　**end while**
22: 　　　　**if** $r = n$ **then**
23: 　　　　　　$r \leftarrow 0$
24: 　　　　　　**while** $r < n$ **do**
25: 　　　　　　　　**if** $\left(\beta_m > 0 \ \& \ \left(\gamma_{n_k}^A + \gamma_{n_k}^Q\right) > 0\right)$ **then**
26: 　　　　　　　　　　Allocate $T_{(i+1)\bmod 2}$ to $Child[m]$
27: 　　　　　　　　　　**decrement** $\beta_m, \gamma_{n_k}^A + \gamma_{n_k}^Q$
28: 　　　　　　　　　　**break loop**
29: 　　　　　　　　**end if**
30: 　　　　　　**end while**
31: 　　　　**end if**
32: 　　**end if**
33: **end for**
34: **if** $\left(\gamma_{n_k}^Q + \gamma_{n_k}^A\right) > 0$ **then**
35: 　　RESOURCE_RELEASE(p_{n_k}, γ_{n_k})
36: **end if**

37: **procedure** RESOURCE_REQUEST($\gamma_{n_k}^Q, \gamma_{n_k}, \alpha^c, \beta^c$)
38: 　　$\gamma_{n_k} \leftarrow \alpha^c + \beta^c$
39: **end procedure**

40: **procedure** RESOURCE_RELEASE($\gamma_{n_k}^Q, \gamma_{n_k}^A, \alpha^c, \beta^c$)
41: 　　Release the remaining $\left(\gamma_{n_k}^Q + \gamma_{n_k}^A\right)$ number of Slots
42: 　　$\gamma_{n_k} \leftarrow (\alpha^c + \beta^c) - \left(\gamma_{n_k}^Q + \gamma_{n_k}^A\right)$
43: **end procedure**

Figure 5 Network configuration for 1-hop scenario with uniform load from both the children nodes.

The RESOURCE_RELEASE procedure releases the excess time slots to its parent for use by the higher level clusters. If the Q-demands of the children nodes cannot be met, the cumulative slot demand ($\alpha^c + \beta^c$) is placed to its immediate parent node for additional slot allocation. This process is carried out by RESOURCE_REQUEST procedure. A parent node allocates Q-share of the children nodes in round-robin fashion. Therefore, the unused slots of a children node are automatically shared among the other needy nodes. Sending additional slot demand to the parent and allocating unused slots in round-robin

fashion implicitly serves the purpose of *resource request* and *resource grant.*

4 Simulation results

4.1 Experimental setup

Performance of the proposed bandwidth allocation scheme has been evaluated through extensive simulations in NS-2 [20] and compared with TreeMAC [7]. TreeMAC is a very similar protocol proposed for wireless sensor networks in which parent node allocates dynamic bandwidth to children nodes according to their traffic demands. The proposed protocol treats QoS-bound and best-effort traffic demands differently whereas TreeMAC does not.

The following metrics have been considered for performance evaluation of our proposed protocol:

(i) Throughput: Throughput refers to the average number of successfully delivered bytes at the destination per second. It is an important metric to provide minimum level of service in a network.

(ii) Delay: It is the time difference between the time a packet was delivered at the destination and it was sent by the source node. Delay is a very essential parameter for delay-sensitive real-time traffic.

Various experiments have been conducted to analyze the impact of the proposed protocol on throughput and delay of both real-time and best-effort traffic. The simulation is carried out for a duration of 300 s. Table 1 presents the different parameters considered along with their values for the simulation purpose.
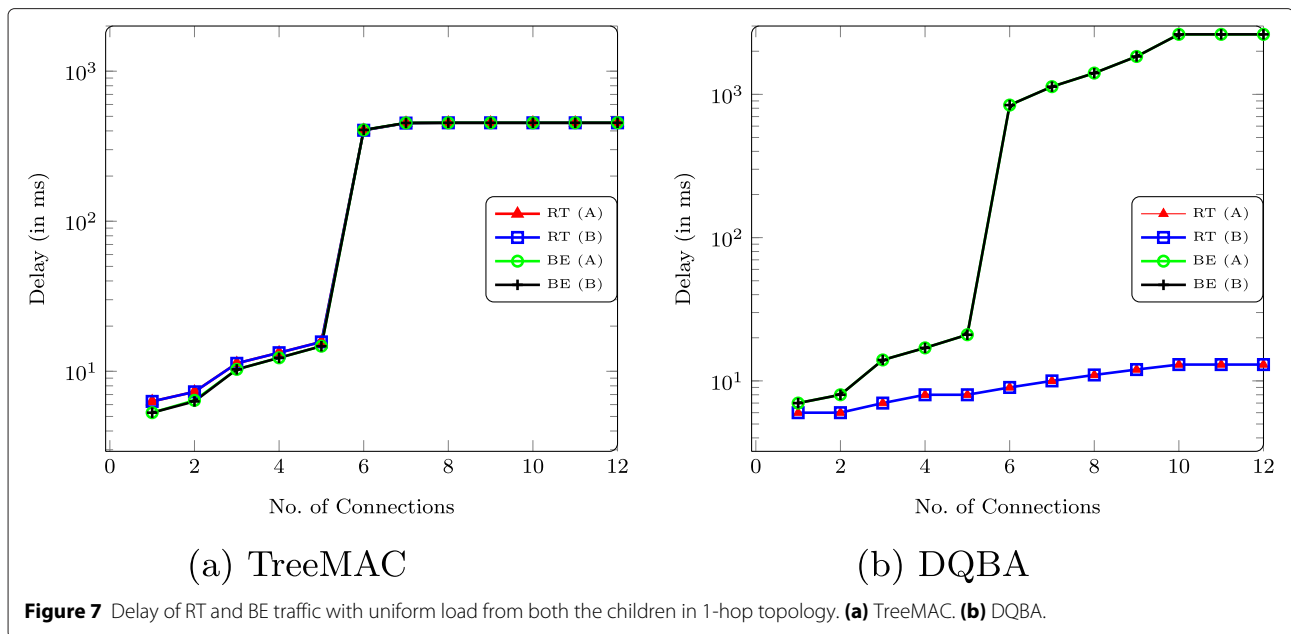

(a) TreeMAC

(b) DQBA

Figure 6 Throughput achieved by RT and BE traffic with uniform load from both the children in 1-hop topology. **(a)** TreeMAC. **(b)** DQBA.

Hussain *et al. EURASIP Journal on Wireless Communications and Networking* (2015) 2015:160

Page 10 of 18
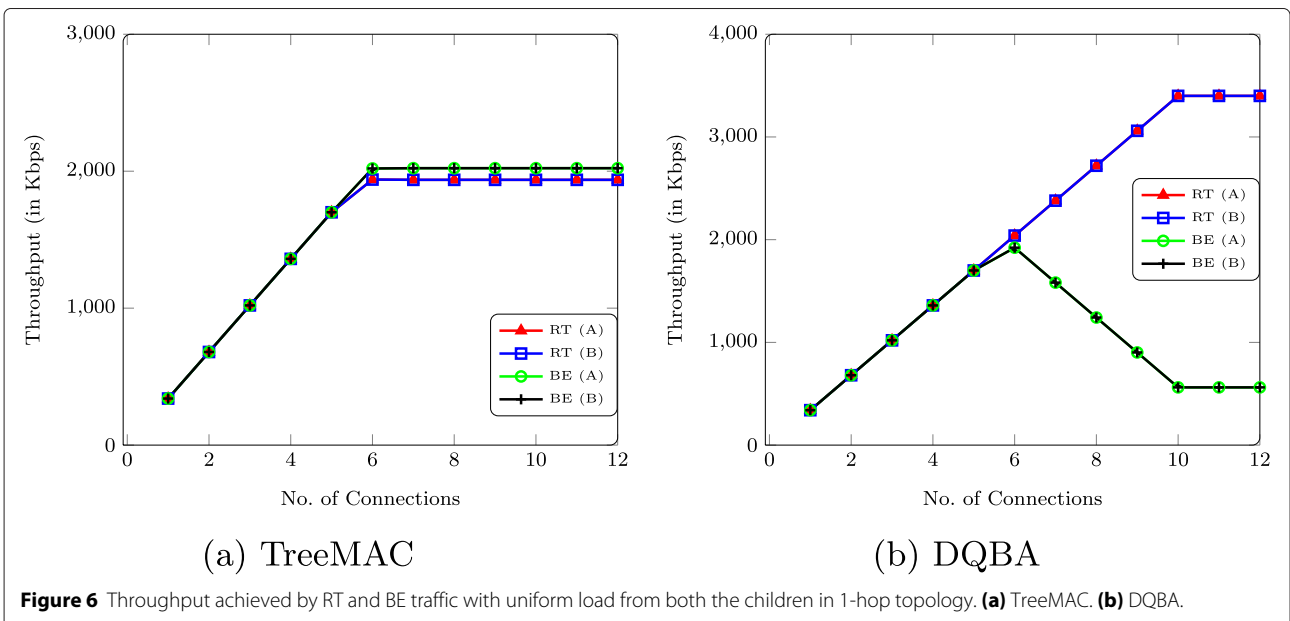
Figure 7 Delay of RT and BE traffic with uniform load from both the children in 1-hop topology. **(a)** TreeMAC. **(b)** DQBA.

Videoconferencing traffic is simulated by using VBR application. VBR and CBR applications have been used to introduce real-time (we call it as Q-demand) and best-effort (we call it as A-demand) traffic, respectively.
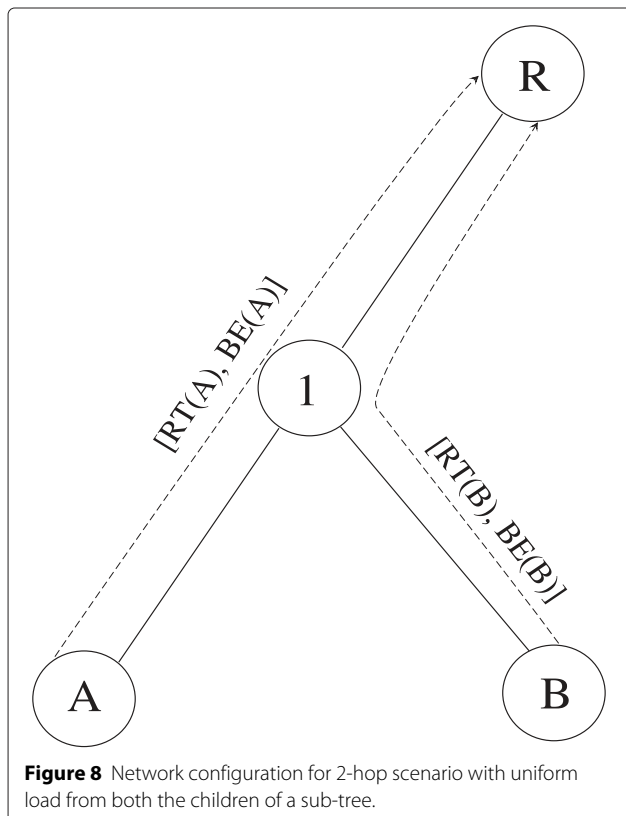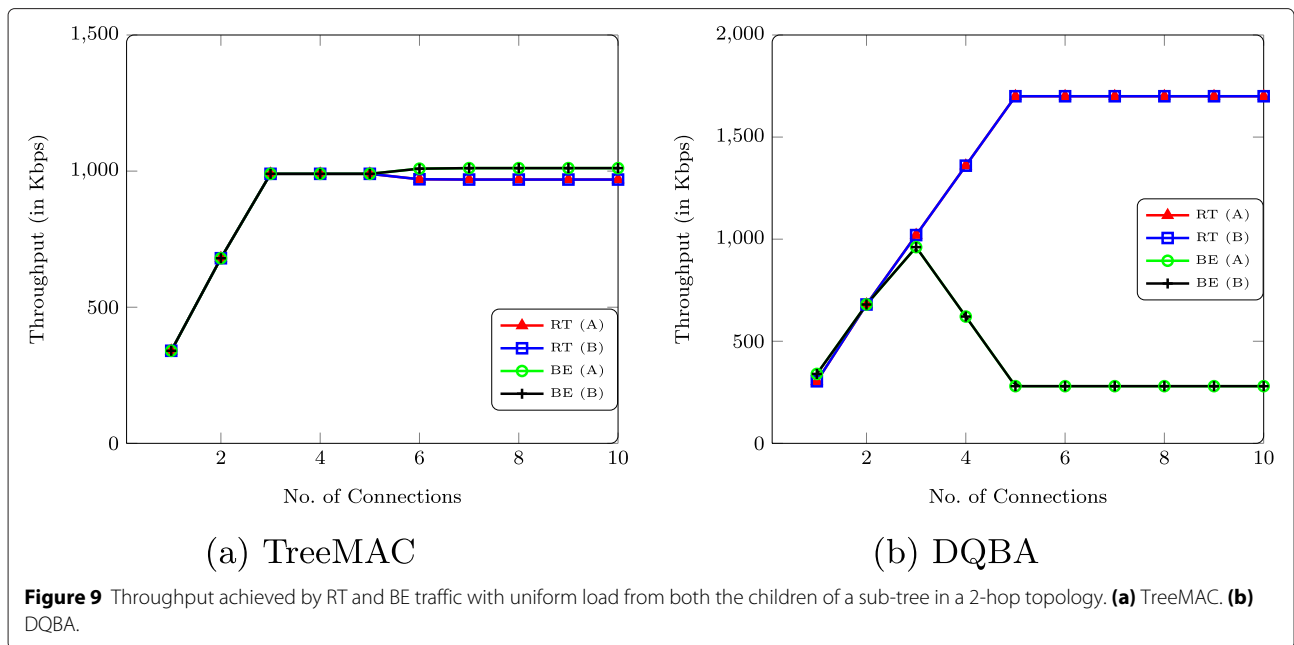

Figure 8 Network configuration for 2-hop scenario with uniform load from both the children of a sub-tree.

A network topology as shown in Figure 4 is considered for the simulation purpose. Node *R* is the root node whereas nodes 1, 2, and 3 are intermediate, and nodes *A*, *B*, and *C* are leaf nodes. From this topology, different cases with 1-hop, 2-hop, and 3-hop network topology are created. We represent the real-time and best-effort traffic from a node *Y* as RT(Y) and BE(Y), respectively. Throughput and delay performance of the proposed scheme is analyzed in two different situations: i) when all the children nodes of a cluster have equal bandwidth demand. This scenario is simulated by offering similar Q-demands and A-demands from both the sub-trees of the root. The demands generated are gradually increased by adding more numbers of CBR connections and ii) when Q-demand is available only in a few children nodes. The other nodes may or may not have A-demand. This scenario is created by offering both Q-demands and A-demands from the left sub-tree and only A-demands from the right sub-tree. In a similar setup, all the above experiments have been carried out using TreeMAC protocol. Finally, the throughput and delay performance of DQBA is compared with TreeMAC.

4.2 Throughput and delay performance

We have evaluated the throughput and delay performance of DQBA and TreeMAC protocols in this section. Traffic load is gradually increased by adding more number of videoconferencing connections (real-time traffic). Videoconferencing is chosen as real-time traffic load as it is both delay and bandwidth sensitive. It introduces packets of size 1,250 bytes at an interval of 33 ms.
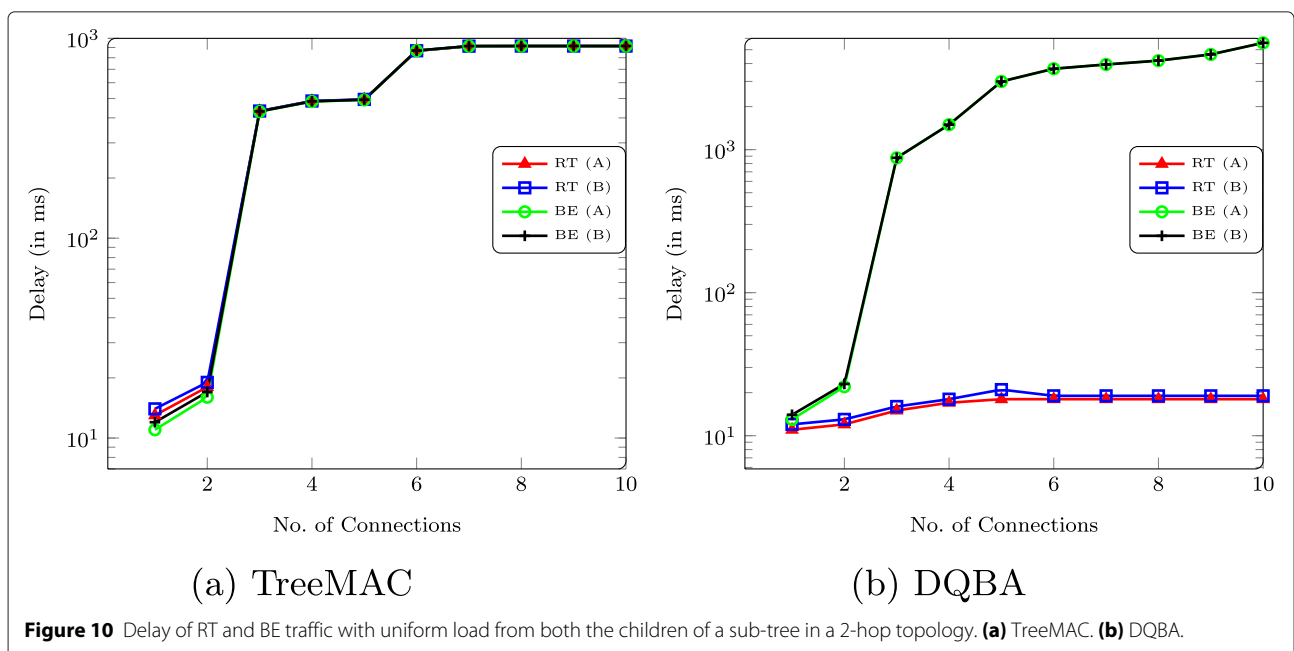
Figure 9 Throughput achieved by RT and BE traffic with uniform load from both the children of a sub-tree in a 2-hop topology. **(a)** TreeMAC. **(b)** DQBA.

The load of best-effort traffic is equally increased with real-time traffic. Since delay performance has varied in a larger range, they are presented in logarithmic scale.

4.2.1 With uniform traffic load

Uniform traffic load situation is created by introducing equal amount of A-traffic and Q-traffic from both left and right sub-trees. Throughput and delay performances in 1-hop, 2-hop, and 3-hop scenarios are analyzed in these traffic situations. Considered simulation topology and the experiment results are discussed in the respective sections of all the considered scenarios.

4.2.2 Throughput and delay performance in 1-hop scenario

The simulation topology and traffic pattern for this experiment is shown in Figure 5. From Figure 6a, it can be
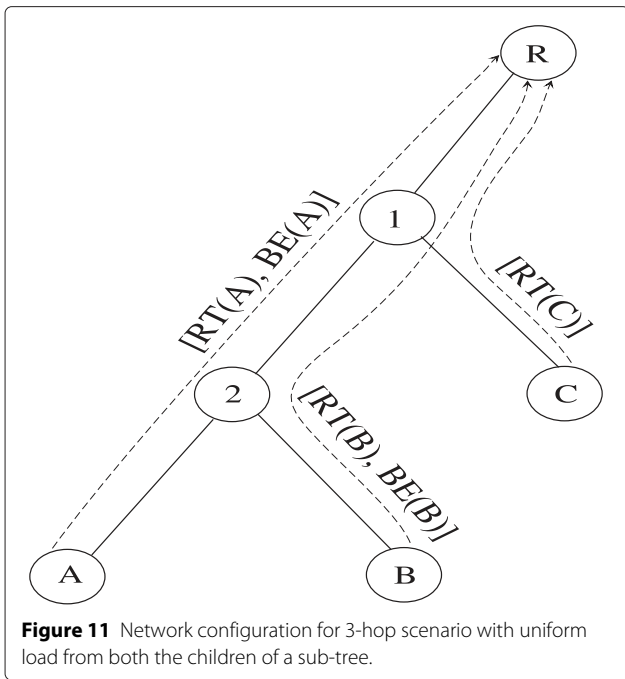

Figure 10 Delay of RT and BE traffic with uniform load from both the children of a sub-tree in a 2-hop topology. **(a)** TreeMAC. **(b)** DQBA.

Hussain *et al. EURASIP Journal on Wireless Communications and Networking* (2015) 2015:160

Page 12 of 18

Figure 11 Network configuration for 3-hop scenario with uniform load from both the children of a sub-tree.

observed that throughput continues to increase till the load reaches the saturation point. Throughput saturation occurs approximately at 4 Mbps of aggregate load. This is true for both TreeMAC as well as DQBA. As the nodes A and B are 1-hop children of R, throughput saturation for the links $A \rightarrow R$ and $B \rightarrow R$ occur at around 4 Mbps aggregate load. Beyond the saturation point, throughput of both types of traffic show invariable performance in

TreeMAC. As our scheme provides priority to Q-demand over A-demand, performance or real-time traffic is not affected till the total bandwidth (3.4 Mbps) is exhausted. It may be noted that a minimum of 20% bandwidth has been kept reserved for best-effort traffic. This is shown in Figure 6b.
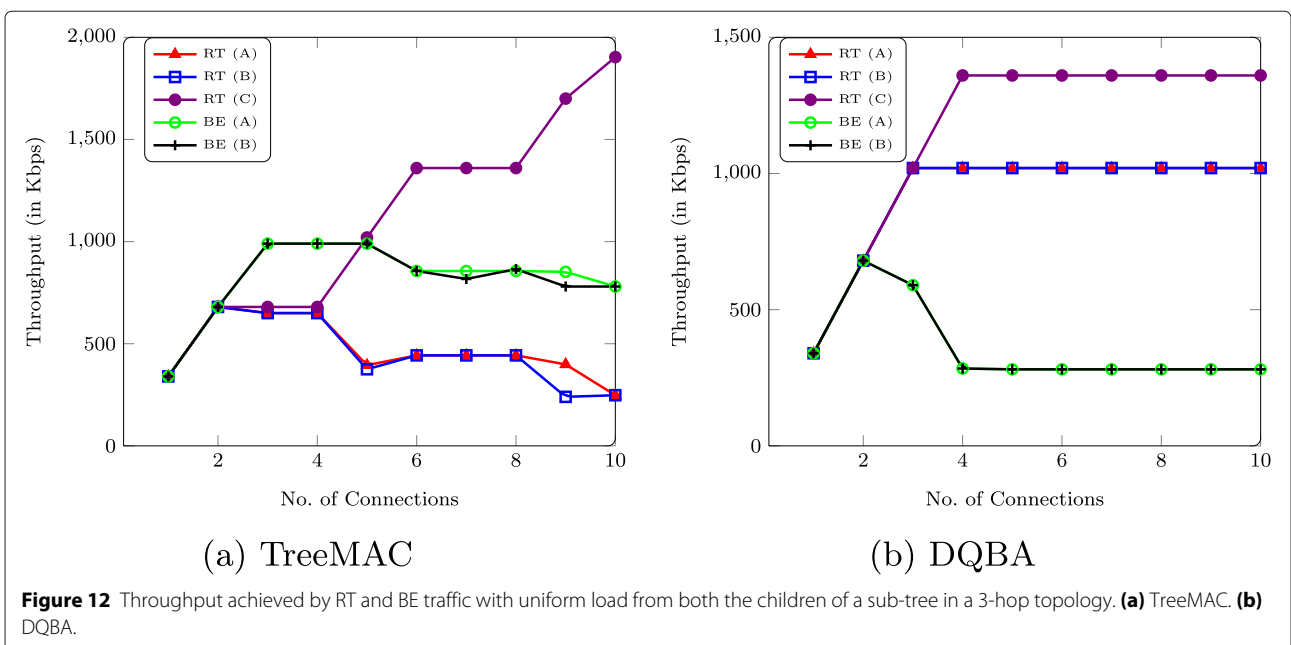
Delay performance of both the protocols have been shown in Figure 7a. In normal offered load, both TreeMAC and DQBA show excellent delay quality. Once the network is saturated with traffic, delay of both RT and BE traffic quickly reach a very high value in TreeMAC (see Figure 7a). DQBA improves delay performance of RT traffic attaining value as low as 13 ms. This is shown in Figure 7a.

4.2.3 Throughput and delay performance in 2-hop scenario

The 2-hop simulation scenario considered in this experiment is presented in Figure 8, and best-effort traffic is introduced from both the leaf nodes A and B. Here, traffic from the nodes A and B shares the bandwidth of the link $1 \rightarrow R$.

As shown in Figure 9a, with increase in the number of connections in TreeMAC, throughput achieved by real-time and best-effort traffic constantly increases in low load. Beyond the saturation point, throughput remains consistent around a certain value. Figure 9b demonstrates an excellent improvement in the throughput of RT traffic. Since DQBA provides priority to RT traffic, it exhibits constant throughput for RT traffic compromising the BE traffic for both the nodes.

The 2-hop delay performances of both the protocols are demonstrated in Figure 10. Up to two connections of BE
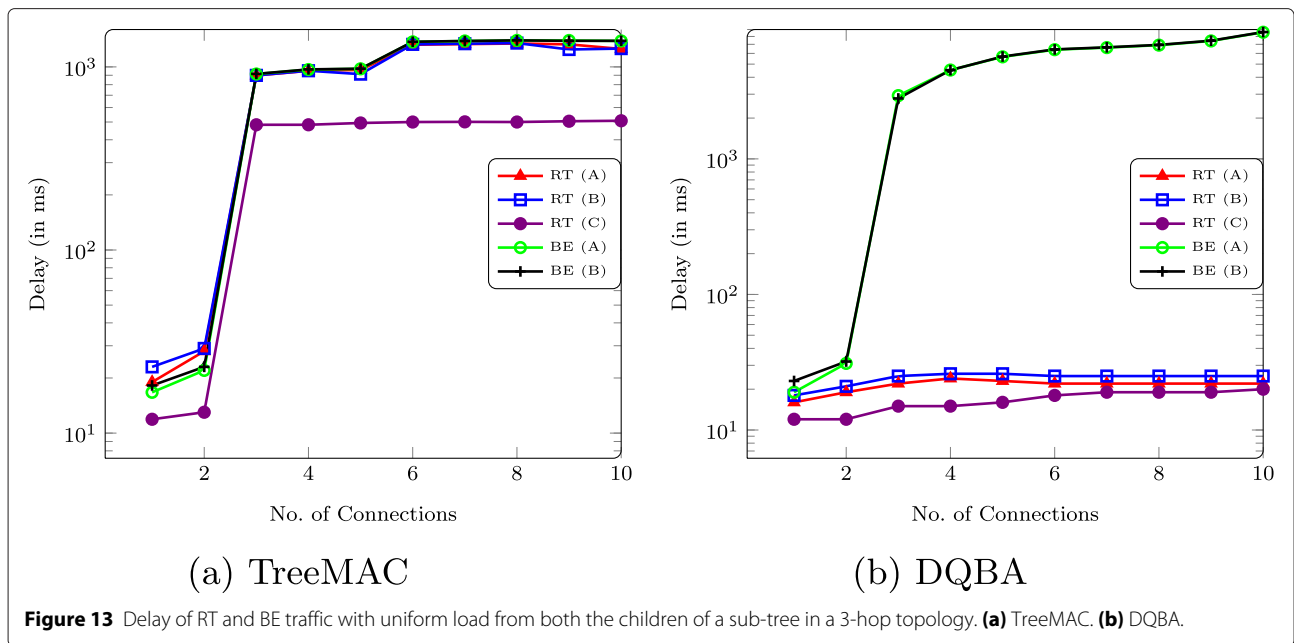

(a) TreeMAC

(b) DQBA

Figure 12 Throughput achieved by RT and BE traffic with uniform load from both the children of a sub-tree in a 3-hop topology. **(a)** TreeMAC. **(b)** DQBA.

Figure 13 Delay of RT and BE traffic with uniform load from both the children of a sub-tree in a 3-hop topology. **(a)** TreeMAC. **(b)** DQBA.

and RT each, TreeMAC displays perfect delay characteristics. But beyond that point, delay value is increased to an unacceptable level for all types of traffic (Figure 10a). Delay performance of RT traffic has been quite improved by our proposed scheme even in very high load situations. This is as shown in Figure 10b.

4.2.4 Throughput and delay performance in 3-hop scenario

To analyze throughput and delay performance in 3-hop configuration, traffic flows are considered as shown in Figure 11. Nodes A and B transmit both RT and BE traffic to the root node R via the relay nodes 1 and 2. Node C

Figure 14 Network configuration for 1-hop scenario with traffic load from single side of a sub-tree.

transmits only RT traffic to node R. In this case, The nodes A and B share the link $2 \to 1$ which in turn shares the link $1 \to R$ with C.

Figure 12a presents the throughput performance of TreeMAC in the traffic scenario shown in Figure 11. The throughput of the link $2 \to 1$ gets saturated with number of connections close to 2 for all individual traffic flows. Once the saturation point is crossed, RT and BE throughput of both A and B nodes remain similar but with diminished value. However, throughput of node C continues to gain as it is positioned one level higher than A and B nodes and hence gets more bandwidth. In DQBA (Figure 12b), throughput of both RT and BE traffic are streamlined even after the saturation point. RT traffic from node C shows better throughput than the others as it gets larger share being a 1-hop node of the root R.

As shown in Figure 13a, 3-hop topology shows similar delay performance to 2-hop. In normal load, both the protocols show outstanding end-to-end delay character. However, as the load goes beyond the saturation point, delay for RT traffic reaches an unacceptable level. In DQBA (Figure 13b), the same has been improved to a great extent by compromising BE traffic performance.

4.2.5 With skewed real-time traffic load

In our proposed protocol, a node needing more time slots for RT traffic can carry the unused time slots of other nodes. To simulate this setting, we have introduced Q-traffic and A-traffic from the nodes belonging to left sub-tree and only A-traffic from the right sub-tree.
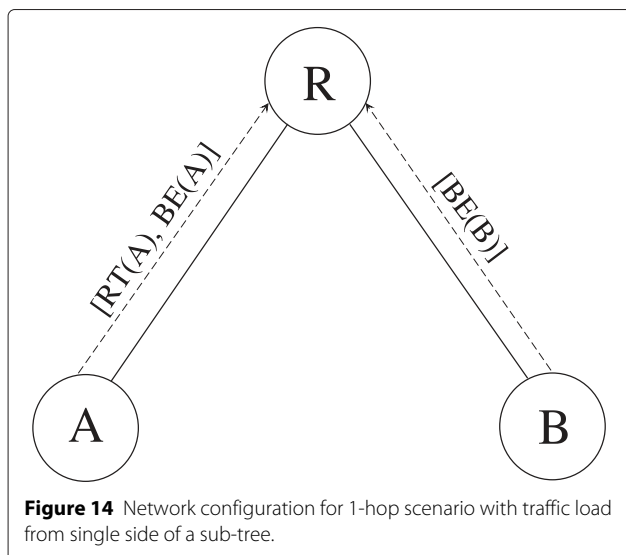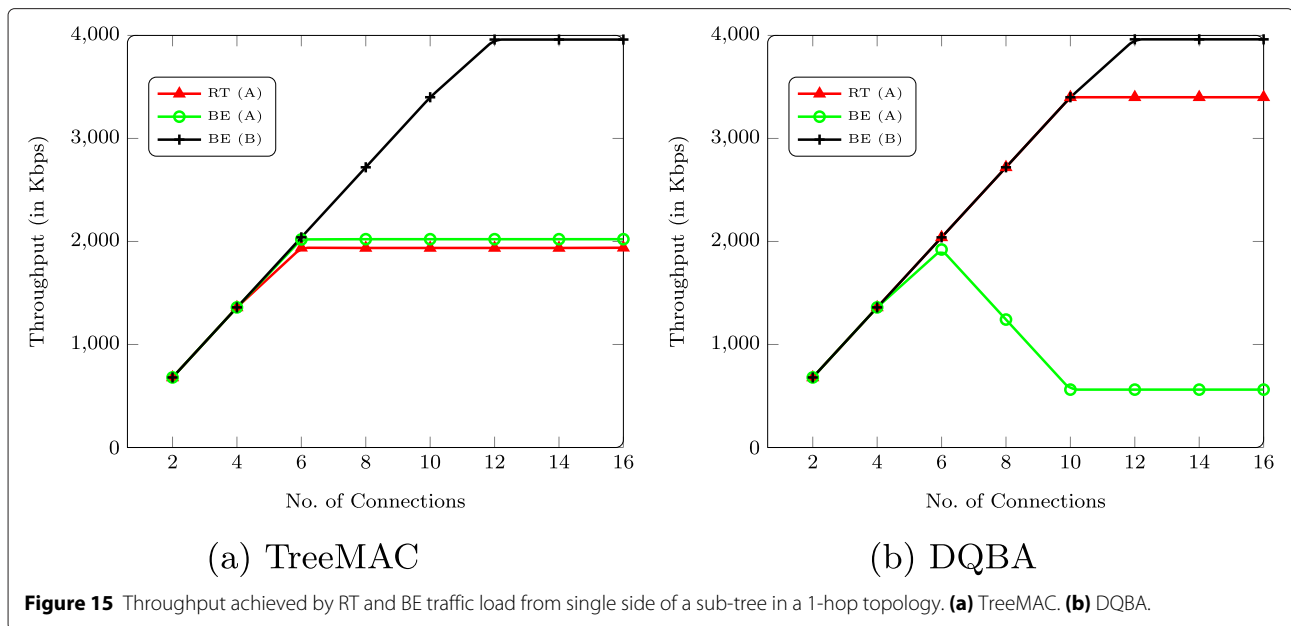
Figure 15 Throughput achieved by RT and BE traffic load from single side of a sub-tree in a 1-hop topology. **(a)** TreeMAC. **(b)** DQBA.

4.2.6 Throughput and delay performance in 1-hop scenario
Figure 14 presents the 1-hop network topology which comprises of two children nodes A and B and a parent node R. Both RT and BE traffic are introduced from A but only A-traffic is given from node B.

Figure 15 presents the throughput performance of TreeMAC and DQBA. The bandwidth of $A \rightarrow R$ link is shared by the RT and BE traffic generated from A. BE traffic originated from B shows higher throughput as it is the only connection using the bandwidth of the link

$B \rightarrow R$. Similar performance is also observed in DQBA (Figure 15b). When the link $A \rightarrow R$ is shared, we see from Figure 15a that the performance of real-time and best-effort traffic originated from the node A remains almost the same in TreeMAC. But with our scheme, the real-time traffic achieves better throughput than best-effort which is shown in Figure 16b. Once the link gets saturated, bandwidth of real-time and best-effort traffic settle according to their maximum allocated bandwidth share.
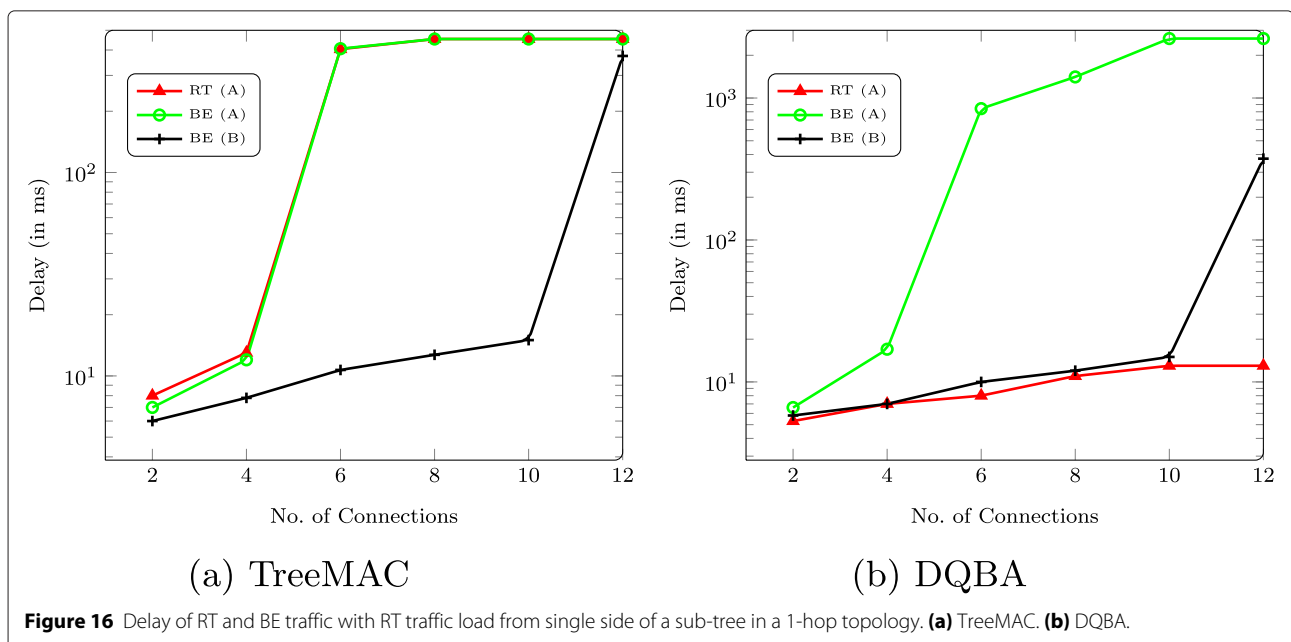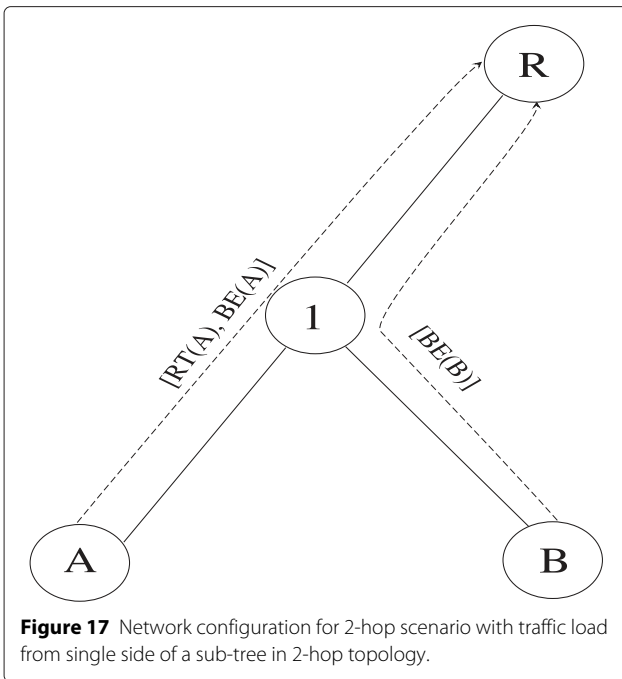

Figure 16 Delay of RT and BE traffic with RT traffic load from single side of a sub-tree in a 1-hop topology. **(a)** TreeMAC. **(b)** DQBA.

Figure 17 Network configuration for 2-hop scenario with traffic load from single side of a sub-tree in 2-hop topology.

In TreeMAC, good delay performance is observed in low load situation as shown in Figure 16a. Beyond the saturation load, delay value quickly moves to a very high level. It establishes the unsuitability of TreeMAC for real-time traffic in multihop WiLD networks. On the other hand, DQBA shows superior real-time traffic delay which is in the order of 13 ms.

4.2.7 Throughput and delay performance in 2-hop scenario
As shown in Figure 17, node *A* transmits both RT and BE

traffic whereas *B* transmits only BE traffic. In this topology and traffic pattern, it is interesting to see how DQBA utilizes the non-utilized time slots of the other sibling nodes.
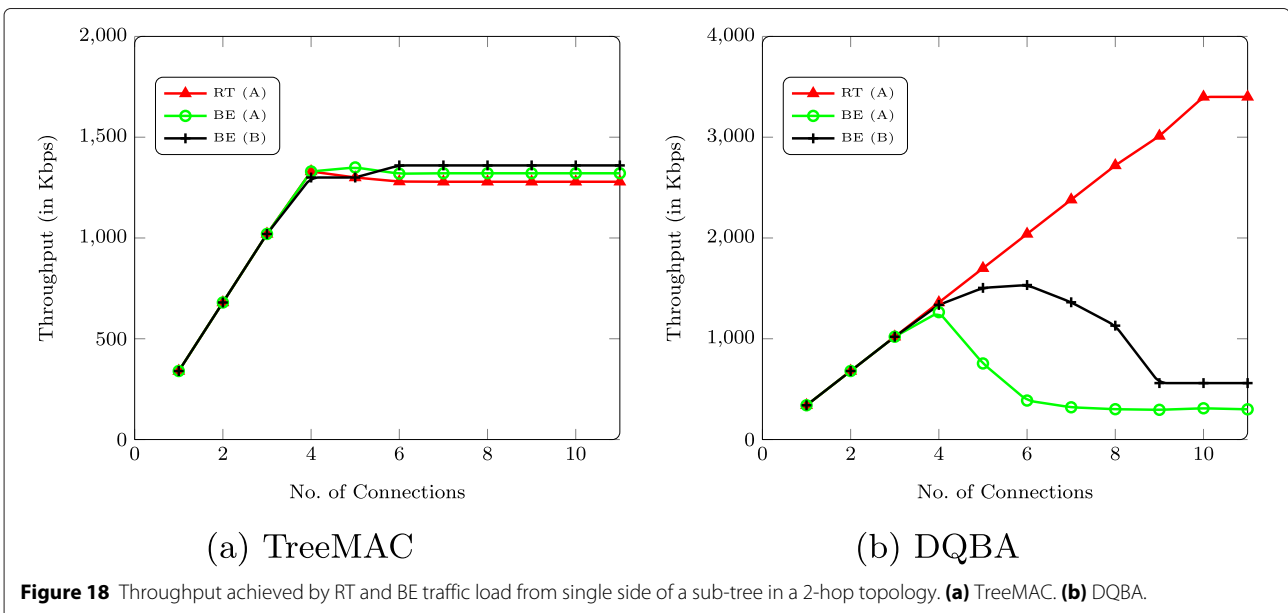
Throughput performance of TreeMAC and DQBA in 2-hop topology under uneven load situation is presented in Figure 18. Throughput performance of TreeMAC as shown in Figure 18a shows a steady increase of throughput in normal load. But in higher load, throughput curbs normally without showing any concern for RT traffic. In this case, the maximum throughput achieved by RT traffic from node *A* is 1,276 Kbps. DQBA improves this figure up to 3,400 Kbps which is shown in Figure 18b. It is exciting to observe that the unused time slots of node *B* are taken away by node *A* in our proposed protocol. A minimum of 20% time slots are kept reserved to avoid node starvation which is normally used by the best-effort traffic.
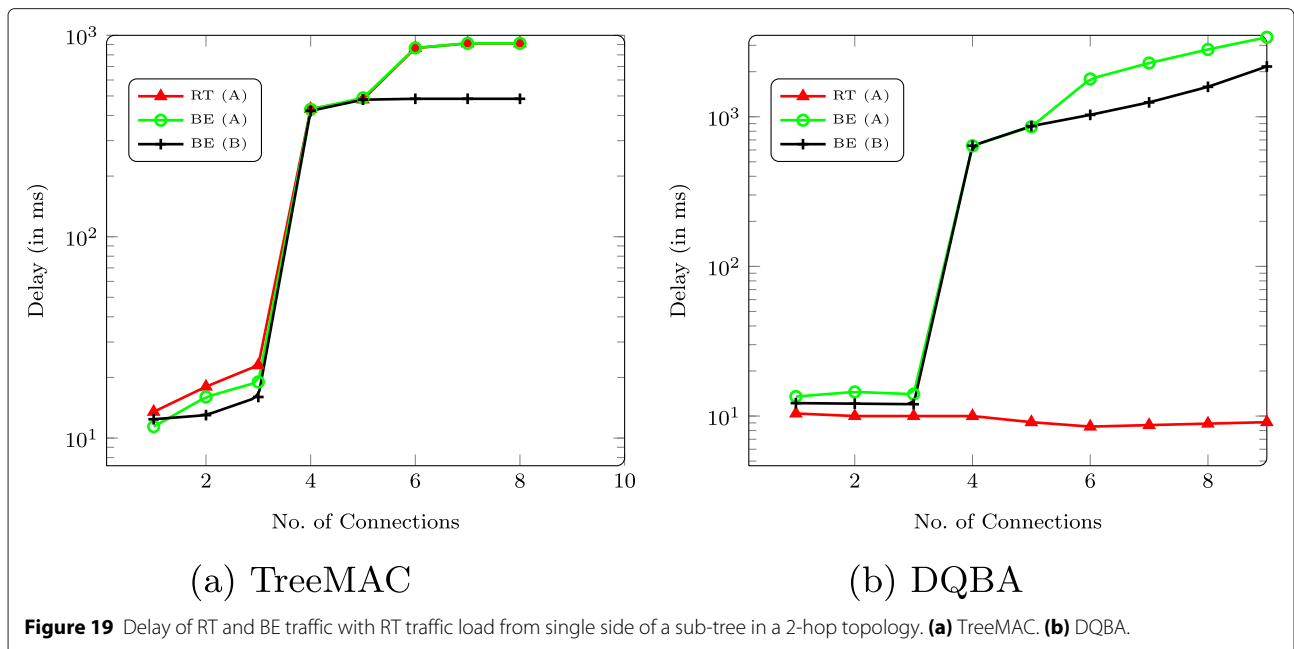
Delay performance of both the protocols are more or less similar to the 1-hop case. DQBA maintains a very small end-to-end delay even in high traffic load. Delay performance are demonstrated in Figure 19.

4.2.8 Throughput and delay performance in 3-hop scenario
In this experiment, we have considered the simulation topology and traffic pattern as shown in Figure 20. RT traffic is introduced only from the node *A* whereas BE traffic is added from all the three leaf nodes, i.e., *A*, *B*, and *C* to *R*.

Throughput performance of TreeMAC in 3-hop topology having single side RT load is similar to 2-hop performance except that the throughput saturation occurs at a lower load (close to 900 Kbps). This is due the hop distance of the source node to the root and link sharing

(a) TreeMAC   (b) DQBA

Figure 18 Throughput achieved by RT and BE traffic load from single side of a sub-tree in a 2-hop topology. **(a)** TreeMAC. **(b)** DQBA.

Hussain *et al. EURASIP Journal on Wireless Communications and Networking* (2015) 2015:160

Page 16 of 18

Figure 19 Delay of RT and BE traffic with RT traffic load from single side of a sub-tree in a 2-hop topology. **(a)** TreeMAC. **(b)** DQBA.

feature of multi-hop WiLD networks. In the proposed protocol, the throughput of real-time traffic increases with the increase in corresponding load whereas the throughput of best-effort traffic diminishes beyond the saturation point (Figure 21b).

Figure 20 Network configuration for 3-hop scenario with traffic load from single side of a sub-tree in a 3-hop topology.

Figure 22 demonstrates the delay characteristics of TreeMAC and DQBA in 3-hop scenario with RT traffic in one of the sub-trees only. Even with a very small number of connections, TreeMAC exhibits very high delay for all types of traffic (Figure 22a). In comparison, as shown in Figure 22b, DQBA provides much better delay performance for real-time traffic in the similar setting.
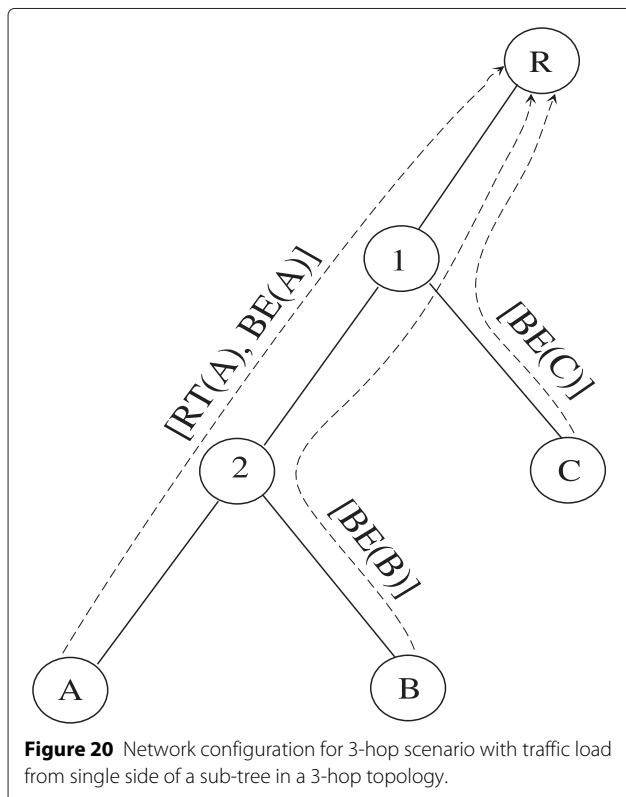
5 Conclusions

In this paper, we have presented a dynamic QoS-aware bandwidth allocation scheme for multi-hop WiLD networks which addresses the congestion problem and hence facilitates QoS support for real-time traffic. The proposed dynamic slot scheduling mechanism efficiently distributes the unused bandwidth among the needy nodes. Twenty percent of the total available slots of each node are not allowed for distribution so as to avoid node starvation problem. Giving higher preference to the real-time traffic, the proposed protocol ensures end-to-end throughput and delay guarantees for the real-time traffic. In skewed load situations, the non-utilized time slots of the nodes are carried over multiple hops easily which provides a feel of using a dedicated link. In the best case, almost two times real-time connections can be supported by the proposed protocol than that of TreeMAC. The simulation results show that the proposed protocol achieves a substantial improvement in throughput and delay of real-time traffic. In this work, we have considered throughput and delay as QoS performance metrics. Uses of other parameters like jitter and packet loss are left as a part of future work.
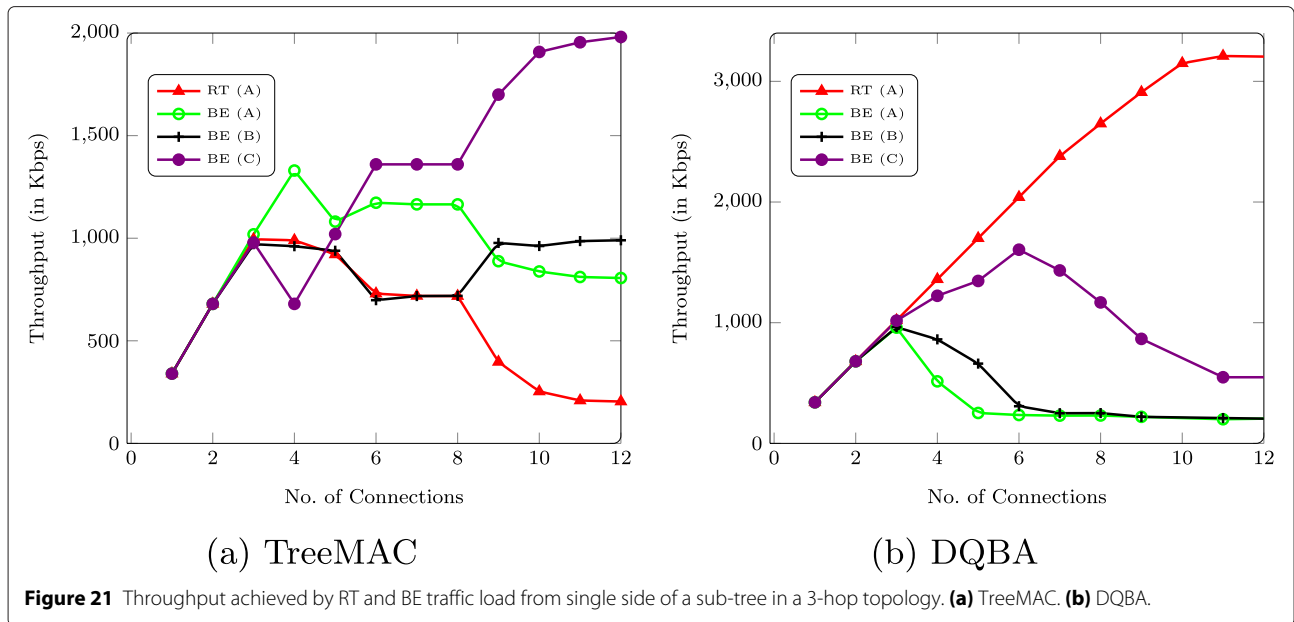
Figure 21 Throughput achieved by RT and BE traffic load from single side of a sub-tree in a 3-hop topology. **(a)** TreeMAC. **(b)** DQBA.
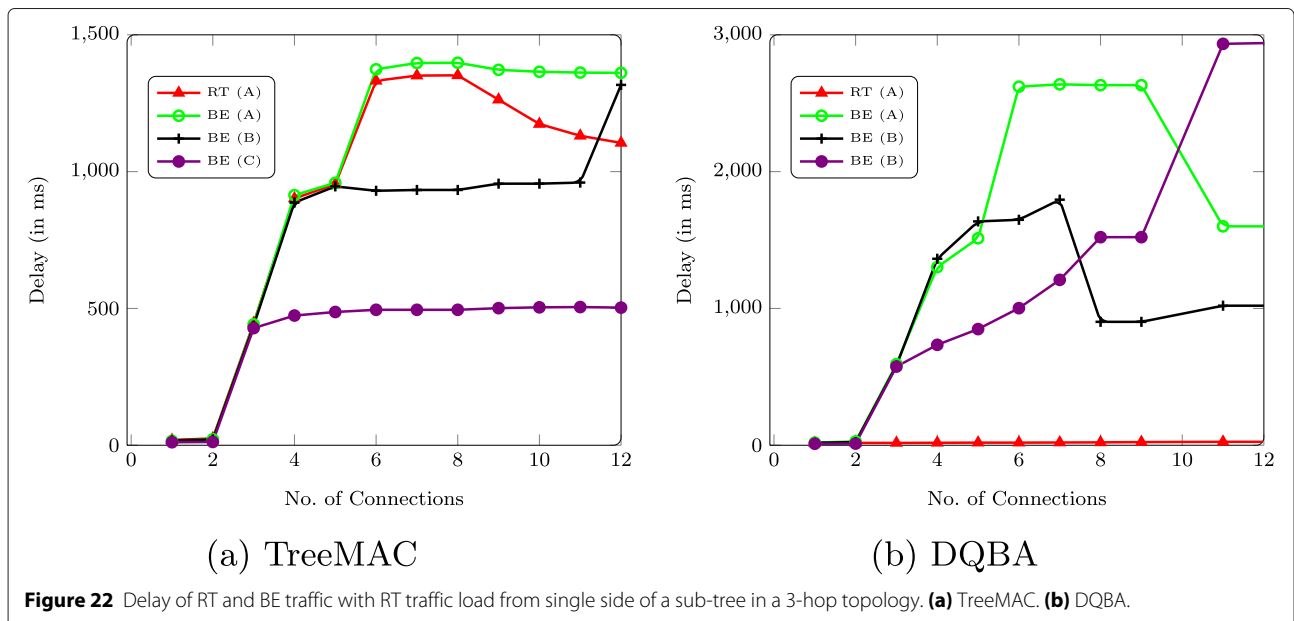

Figure 22 Delay of RT and BE traffic with RT traffic load from single side of a sub-tree in a 3-hop topology. **(a)** TreeMAC. **(b)** DQBA.

Hussain *et al. EURASIP Journal on Wireless Communications and Networking* (2015) 2015:160

Page 18 of 18

Competing interests

We declare that we do not have an competing interests with any other authors involved in this category of research. Any resemblance to actual persons living or dead is purely co-incidental.

Authors' contributions

In this paper, IH and ZIA are involved in literature survey, design, and implementation whereas DKS and NS supervised the work. All authors read and approved the final manuscript.

Author details

[1]Department of Information Technology, North-Eastern Hill University, Umshing, Shillong 793022, Meghalaya, India. [2]Department of Computer Science and Engineering, National Institute of Technology, Laitumkhrah, Shillong 793003, Meghalaya, India. [3]Department of Computer Science and Engineering, Tezpur University, Napaam, Tezpur, 784028, Assam, India.

References

1. K Chebrolu, B Raman, in *Proceedings of the 2007 Workshop on Networked Systems for Developing Regions. (NSDR)*. FRACTEL: A fresh perspective on (rural) mesh networks, (2007), pp. 8:1–8:6. http://doi.acm.org/10.1145/1326571.1326583
2. MA El-Gendy, A Bose, KG Shin, Evolution of the Internet QoS and support for soft real-time applications. Proc. IEEE. **91**(7), 1086–1104 (2003)
3. B Raman, K Chebrolu, in *Proceedings of the 11th Annual International Conference on Mobile Computing and Networking. (MobiCom)*. Design and evaluation of a new MAC protocol for long-distance 802.11 mesh networks, (2005), pp. 156–169. http://doi.acm.org/10.1145/1080829.1080847
4. R Patra, S Nedevschi, S Surana, A Sheth, L Subramanian, E Brewer, in *Proceeding of the 4th USENIX Symposium on Networked Systems Design and Implementation (NSDI)*. WiLDNet: Design and implementation of high performance WiFi based long distance networks (ACM, 2007)
5. V Gabale, B Raman, P Dutta, S Kalyanraman, A classification framework for scheduling algorithms in wireless mesh networks. Commun. Surv. Tutorials IEEE. **15**(1), 199–222 (2013)
6. S Nedevschi, RK Patra, S Surana, S Ratnasamy, L Subramanian, E Brewer, in *Proceedings of the 14th ACM International Conference on Mobile Computing and Networking. (MobiCom)*. An adaptive, high performance MAC for long-distance multihop wireless networks, (2008), pp. 259–270. http://doi.acm.org/10.1145/1409944.1409974
7. W-Z Song, R Huang, B Shirazi, R LaHusen, TreeMAC: Localized TDMA MAC protocol for real-time high-data-rate sensor networks. Pervasive Mobile Comput. **5**(6), 750–765 (2009)
8. I Rhee, A Warrier, M Aia, J Min, ML Sichitiu, Z-MAC: a Hybrid MAC for wireless sensor networks. IEEE/ACM Trans. Netw. (TON). **16**(3), 511–524 (2008)
9. G-S Ahn, SG Hong, E Miluzzo, AT Campbell, F Cuomo, in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems. (SenSys)*. Funneling-mac: A localized, sink-oriented MAC for boosting fidelity in sensor networks, (2006), pp. 293–306. http://doi.acm.org/10.1145/1182807.1182837
10. S Zhuo, Y-Q Song, Z Wang, Z Wang, in *9th IEEE International Workshop on Factory Communication Systems (WFCS)*. Queue-MAC: A queue-length aware hybrid CSMA/TDMA MAC protocol for providing dynamic adaptation to traffic and duty-cycle variation in wireless sensor networks, (2012), pp. 105–114
11. I-S Hwang, J-Y Lee, C-W Huang, Z-D Shyu, in *10th International Conference on Mobile Data Management: Systems, Services and Middleware*. Advanced dynamic bandwidth allocation and scheduling scheme for the integrated architecture of EPON and WiMAX (IEEE, 2009), pp. 655–660
12. Y Ben-David, M Vallentin, S Fowler, E Brewer, in *Proceedings of the 4th ACM Workshop on Networked Systems for Developing Regions. (NSDR)*. Jaldimac: Taking the distance further, (2010), pp. 2:1–2:6. http://doi.acm.org/10.1145/1836001.1836003
13. V Sevani, B Raman, P Joshi, Implementation-based evaluation of a full-fledged multihop TDMA-MAC for WiFi mesh networks. IEEE Trans. Mobile Comput. **13**(2), 392–406 (2014)
14. V Gabale, B Raman, K Chebrolu, P Kulkarni, in *Proceedings of the First ACM Symposium on Computing for Development. (ACM DEV)*. LiT MAC: Addressing the challenges of effective voice communication in a low cost, low power wireless mesh network (ACM New York, NY, USA, 2010), pp. 5:1–5:11. http://doi.acm.org/10.1145/1926180.1926187
15. G-S Ahn, SG Hong, E Miluzzo, AT Campbell, F Cuomo, in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems. (SenSys)*. Funneling-MAC: a localized, sink-oriented MAC for boosting fidelity in sensor networks (ACM New York, NY, USA, 2006), pp. 293–306
16. I Slama, B Shrestha, B Jouaber, D Zeghlache, in *33rd IEEE Conference on Local Computer Networks (LCN)*. A hybrid MAC with prioritization for wireless sensor networks, (2008), pp. 274–281
17. S Zhuo, Z Wang, Y-Q Song, Z Wang, L Almeida, in *10th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*. iQueue-MAC: A traffic adaptive duty-cycled MAC protocol with dynamic slot allocation (IEEE, 2013), pp. 95–103
18. K Miray, K Ibrahim, K Ezhan, Utilization-based dynamic scheduling algorithm for wireless mesh networks. EURASIP J. Wireless Commun. Netw. (2010)
19. B Paul, On distributed communications. Volumes I-XI, RAND Corporation Research Documents, 637–648 (1964)
20. The Network Simulator - ns-2. http://www.isi.edu/nsnam/ns/