HQMR: Hybrid QoS based Routing Protocol for Wireless Mesh Environment

Hajer Bargaoui, Nader Mbarek, Olivier Togni
LE2I Laboratory, University of Burgundy
Dijon, France
e-mails : {Hajer.Bargaoui, Nader.Mbarek, Olivier.Togni}@u-bourgogne.fr

Mounir Frikha
MEDIATRON Laboratory, High School of Communication of Tunis (SUP'COM)
Tunis, Tunisia
e-mail : m.frikha@supcom.rnu.tn

***Abstract*—Wireless Mesh Networks (WMNs) have been attracting more and more interest from both academic and industrial environments for their seamless broadband connectivity to Internet networks. Besides, providing QoS guarantees for real-time and streaming applications such as Voice over IP (VoIP) and Video on Demand (VoD) is a challenging issue in such environment. Thus, we propose a novel QoS based routing protocol for wireless mesh infrastructure, called Hybrid QoS Mesh Routing (HQMR). Moreover, a clustering algorithm is developed to enhance scalability issues within the mesh infrastructure. HQMR is composed of two routing sub-protocols: a reactive routing protocol for intra-infrastructure communications and a proactive QoS based multi-tree routing protocol for communications with external networks. The proposed routing protocol ensures forwarding real-time and streaming applications with QoS guarantee in a mesh wireless environment. We analyze in this paper the simulation results of different scenarios conducted on the network simulator ns-3 to demonstrate the effectiveness of the reactive routing sub-protocol in forwarding real-time applications with QoS guarantee.**

Keywords-Wireless Mesh Network; QoS routing; HQMR; Network Simulator; ns-3.

I. INTRODUCTION

Recently, wireless mesh networks have received increased attention from researchers and industrial environments [1]. They have emerged as a key wireless technology for numerous applications such as broadband home networking, community and neighborhood networks, enterprise networking, etc. [2], [3]. Besides, they are a promising solution to provide last-mile connectivity to Internet for fixed and/or mobile users in zones where wired networks deployment is difficult. These abilities are provided thanks to their various qualities. They are self-organizing and self-configuring networks where participating nodes automatically establish and maintain connectivity. They enable also quick deployment, easy maintenance, low cost, high scalability, etc. These benefits have motivated consistently researchers to study their characteristics for better performance.

In fact, a wireless mesh network is composed of two types of node: wireless mesh routers and wired/wireless mesh clients. The mesh routers are static and non-power constrained nodes and the mesh clients are potentially mobile nodes. In fact, mesh routers communicate between each other in multi-hop fashion, forming a relatively stable network and the mesh clients are connected to these routers using a wireless or a wired link. The role of most mesh routers in a wireless mesh network is to perform relaying of data for other mesh routers, a typical ad-hoc networking paradigm. Some other mesh routers have also additional gateway capabilities. These nodes, named mesh gateways, enable the integration of wireless mesh networks with various other networks and often have a wired link to Internet, helping in forwarding clients traffic and in providing Internet services to the mesh clients.

One major challenge for wireless mesh networks is to provide QoS support. Since deployments of WMNs continue to grow, providing Quality of Service for real-time and streaming applications, such as VoIP and VoD, is an important task. Moreover, establishing paths with the highest performance is a challenging issue for routing protocols within wireless mesh networks in order to satisfy applications' requirements.

However, the different research works proposing routing solutions on wireless mesh networks rely simply on adapting protocols originally designed for mobile ad hoc networks and adding a little support for QoS. In this paper, we propose a hybrid QoS based routing protocol, called Hybrid QoS Mesh Routing (HQMR) [1], which exploits more efficiently the particular topology of a wireless mesh network, based on a hybrid wireless mesh architecture. The proposed wireless mesh architecture is formed by an IEEE 802.16j based infrastructure and different IEEE 802.11s based client domains. Furthermore, in order to solve scalability issues and reduce efficiently the network's load, a clustering algorithm is proposed for the IEEE 802.16j infrastructure of our global wireless mesh architecture. HQMR is then deployed on the IEEE 802.16j infrastructure to ensure routing functionalities. It is a hybrid protocol adopting a reactive routing sub-protocol for intra-infrastructure communications and a proactive multipath tree-based routing sub-protocol for inter-infrastructure communications, where the mesh gateway is considered as a root.

The remainder of this paper is organized as follows. In Section II, we present two standards of wireless mesh networks. Related works are presented in Section III. Section IV introduces the architecture of our framework. Then, the proposed HQMR routing protocol is defined in Section V. Section VI defines two usage scenarios of HQMR to illustrate its processing. We introduce respectively, the performance evaluation of IMRR routing sub-protocol and the results analysis in Section VII and Section VIII. Finally, Section IX concludes the paper.

II. STATE OF THE ART

Given the increased interest in wireless mesh networks, different standards have been specified. In this section, we present the latest standardization results namely IEEE 802.11s standard based on Wi-Fi technology and IEEE 802.16j standard built on the WiMAX technology.

A. *IEEE 802.11s Standard*

The IEEE 802.11s standard started initially as a study group in 2003, and became a Task Group in July 2004 for developing a flexible and extensible solution for wireless mesh networks based on IEEE 802.11 technology. The first draft was accepted in March 2006 and their work was approved by 2011 [4].

An IEEE 802.11s network is formed by a wireless infrastructure, composed of a set of mesh routers named Mesh Points (MP), to which the mesh clients (STA) are connected to access the Internet services. Some MPs, named Mesh Access Point (MAP), have additional access point functionalities to help connecting the STA nodes to the mesh infrastructure. Other MPs named Mesh Portal Point (MPP) have gateway functionalities to ensure the connection between the mesh cloud and the external network.

Besides, IEEE 802.11s standard defines a layer 2 basic routing protocol named HWMP (Hybrid Wireless Mesh Protocol) and, therefore, uses MAC addresses and a radio routing metric.

B. *IEEE 802.16j Standard*

IEEE 802.16j task group was officially established in March 2006 and their work was published in 2009. The IEEE 802.16j standard [5], is an amendment to the IEEE 802.16e [6] standard in order to introduce Mobile Multi-hop Relay (MMR) specifications where traffic between a Multi-Relay Base Station (MR-BS) and a Subscriber Station (SS) can be relayed through nodes named Relay Stations (RS). The number of hops between MR-BS and SS is not defined but it must only contain RS nodes.

In fact, IEEE 802.16j has defined two different relay modes: transparent mode and non-transparent mode. In transparent mode, the RS is used to improve the network capacity. It does not forward any signaling frame. It relays only data traffic, that is why the SS, which is physically attached to it, does not know the existence of the RS. The non-transparent mode is usually used to extend the network coverage. The RS nodes in this mode are able to generate their own signaling frame or forward those provided by the MR-BS depending on the scheduling mechanism.

Just like the previous Wimax standard namely IEEE 802.16e [6], IEEE 802.16j manages also the quality of service at the MAC sub-layer by differentiating five service classes, from high to low priority: Unsolicited Grant Service (UGS), real-time Polling Service (rtPS), enhanced real-time Polling Service (ertPS), non real-time Polling Service (nrtPS) and Best Effort (BE).

III. RELATED WORK

A. *QoS Routing*

QoS provisioning is an important issue for wireless mesh networks since they are typically used for providing broadband wireless Internet access to a large number of users and networks. To meet applications' QoS requirements, different QoS routing protocols were proposed for WMNs.

Wireless Mesh Routing (WMR) [7] is a QoS solution for wireless mesh LAN networks. It provides QoS guarantees in terms of minimum bandwidth and maximum end-to-end delay. These two parameters are verified jointly with the route discovery process. The value of the node's available bandwidth is estimated thanks to the bandwidth already in use by the considered node and by its neighboring nodes. Then, the end-to-end delay is estimated by using the round trip delay method [8]. Kon et al. [9] improve the WMR protocol by proposing a novel end-to-end packet delay estimation mechanism with a stability-aware routing policy. The delay estimation is based on packets named DUMMY-RREP, which have the same size, priority and data rate as real data traffic. The robustness of a link is estimated by measuring the number of Hello packets received during a given time.

Some other works include the QoS verification in the route discovery phase. For example, QoS AODV (QAODV) [10] integrates a new metric for IEEE 802.11 mesh networks, composed of bandwidth, delay, hop count and load ratio. In the same way, Rate-Aware AODV (R-AODV) [11] uses minimum network layer transmission time as a performance metric in multi-rate WiFi mesh networks. Mesh Admission control and QoS Routing with Interference Awareness (MARIA) [12] is another QoS aware routing protocol for wireless mesh networks. It is a reactive protocol incorporating an interference model in the route discovery process. This protocol uses a conflict graph model to characterize both inter and intra-flow interference. The available residual bandwidth is computed based on the maximal clique constraints in its local conflict graph to make distributed hop-by-hop admission control decision.

In this context, we propose the HQMR protocol to provide QoS provisioning routing functionalities within a wireless mesh environment.

B. *Clustering*

Clustering concept was introduced to organize large wireless multi-hop networks into groups named clusters. Every cluster is coordinated by a cluster-head to achieve basic network performances, even with mobility and limited energy resources. The different clustering algorithms differ mainly in the method used for the election of the cluster-heads: Lowest-ID heuristic [13], Highest-degree heuristic [14] and node-Weight heuristic [15]. The Lowest-ID algorithm [13] designs the node with the lowest-ID as cluster-head. Then, a cluster is formed by that node and all its neighbors. In order to maintain inter-clusters connectivity, Gateway-nodes are defined. The Highest-degree algorithm [14] uses the degree of the node (number of the neighbors) for cluster-head election process. The third type of clustering algorithms calculates a weight for each node according to specific metrics. For example, the authors in [15] propose an algorithm that takes into consideration the number of nodes a cluster-head can handle ideally without any severe degradation in the network performances, transmission power, mobility and battery power of the node.

Combining clustering algorithms with routing protocols offers better performances within the network layer, by reducing the amount of control messages propagated inside the network since the exchange is limited within a cluster; and by minimizing the size of routing tables at each node since it stores only the information of its cluster.

Zone Routing Protocol (ZRP) [16] is a cluster-based routing protocol for ad hoc networks that uses different routing sub-protocols for inter and intra-clusters communications. Within a cluster zone, a proactive component is used to maintain up-to-date routing tables. Routes outside the routing zone are explored with a reactive component combined with a border-casting concept. This concept utilizes the topology information provided by the proactive protocol to direct query requests to the nodes in the border of the zone.

Singh et al. [17] propose a hierarchical cluster based routing protocol for wireless mesh networks, in which the mesh gateway is the highest level node. When a node has a data packet to forward, it sends a path request message to its cluster-head. In case that the destination is not in the same group, the cluster-head sends the path request message to the mesh gateway, which forwards the request to the other cluster-heads. Similarly, the research work in [18] defines a multi-level clustering approach with a reactive routing protocol for wireless mesh networks, in order to reduce the load on the mesh gateway. The source node unicasts the route request message to its cluster-head. If a route is not found, the cluster-head forwards the message to an upper-level node, firstly to a Group Head and then to the mesh gateway. This approach reduces considerably the number of broadcast messages used for route discovery process.

For its benefits, we adopt this concept while adapting it to cluster based routing for our HQMR protocol to solve scalability issues and to offer better routing performances within the wireless mesh infrastructure.

C. *Multipath Routing*

Multipath routing is the technique of using multiple paths between each node pair instead of having a single path, which helps to improve the available bandwidth, to reduce end-to-end delay and to enhance load balancing and fault tolerance [19]. These multiple paths may be used as backup paths or as concurrent paths. The backup paths are used only when the primary path is broken. The concurrent paths are used simultaneously to forward traffic, according to specific traffic distribution mechanism over the used paths [19], [20].

Providing broadband wireless Internet access to end users is an important objective of wireless mesh networks. Thus, most of the traffic is directed either from or towards the Internet mesh gateway. Consequently, some nodes or links could be overloaded since each node will aim to choose the best path to the gateway. The Multi-path Mesh (MMESH) [21] protocol was proposed as a possible solution. It is a proactive multi-path routing protocol, specifying an algorithm to split traffic over multiple selected paths between each node and the mesh gateway, for balancing network load uniformly. Then, MMESH applies a congestion aware approach to choose the best path. Multi-path routing protocols help to improve the performances of a network by using multiple disjoint paths. However, when all these paths are utilized simultaneously to transmit data, they will affect each other by causing route-coupling problem [22]. AODV Decoupled Multipath (AODV-DM) [23] was developed to establish efficiently node-disjoint paths that are enough separated to avoid inter-path interferences. It selects the primary path according to the single path selection process. Then, a region is defined around this primary path, so that the second path will be selected outside it. Thereby, these two paths are not only disjoint, but also decoupled paths.

Multi-path Hybrid Routing Protocol (MHRP) [24] is a multipath routing protocol used with a hybrid architecture formed by a mesh infrastructure and ad hoc client domains. This protocol is based on the backup routes concept to enhance the network performance.

Considering the importance of the traffic from/towards the mesh gateway in a wireless mesh network, we propose the integration of the multipath routing concept as a promising solution for network load balancing in our wireless mesh infrastructure.

IV. PROPOSED GLOBAL HYBRID WIRELESS MESH ARCHITECTURE

For our framework, we adopt a hybrid wireless mesh network architecture, combining two different technologies. It is formed by a non-transparent IEEE 802.16j-based infrastructure and IEEE 802.11s-based client domains (Fig. 1). A hybrid QoS based routing protocol (HQMR) is also proposed within the wireless mesh infrastructure.

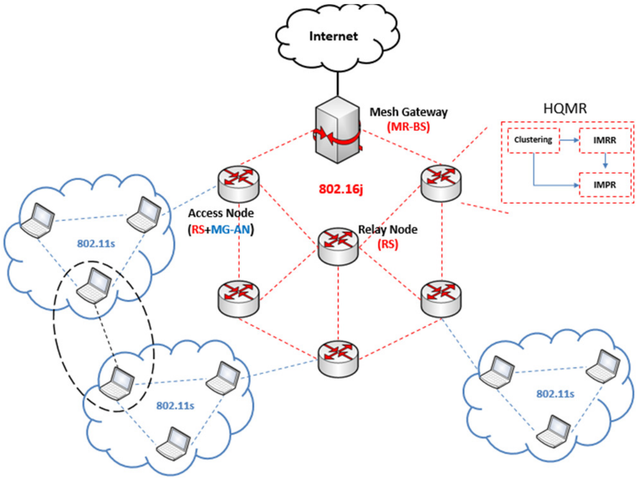

Figure 1. Global hybrid wireless mesh architecture

A. *IEEE 802.16j-based mesh infrastructure domain*

For the wireless mesh infrastructure, we use the non-transparent relay mode of the IEEE 802.16j technology to ensure a better coverage. Then, in order to organize the functionalities of each node, we define three types of nodes within the mesh infrastructure: the Mesh Gateway (MG), the Relay Nodes (RN) and the Access Nodes (AN). The MG is the intermediate node between the Internet cloud and the wireless mesh infrastructure. It helps forwarding clients requests to the Internet network. The RNs are the nodes located in the core of the mesh infrastructure to ensure forwarding traffic flows from a node to another inside it. Last, we consider the nodes located in the border of the infrastructure, as Access Nodes (AN). They provide interconnection between the mesh infrastructure and the client domains. Thus, compared to the topology of an IEEE 802.16j network, our MG and RN nodes have, respectively, the same functionalities as the MR-BS node and the RS nodes. In fact, the AN nodes may be considered as bridge nodes playing both the role of a relay node in the IEEE 802.16j infrastructure and the role of a gateway in the IEEE 802.11s area. Thus, they are equipped with two radio interfaces: one is operating with the WiMAX technology [6] and another with Wi-Fi technology [25].

At each relay node of the wireless mesh infrastructure (including the ANs and the MG), the proposed routing protocol (HQMR) must be implemented with our clustering algorithm to reduce mainly the size of the routing tables. The different blocks of HQMR will be described in Section IV.

B. IEEE 802.11s-based mesh client domain

The client domains are formed by a set of IEEE 802.11s [4] MP (Mesh Point), which are interconnected to each other forming the mesh topology and by a gateway node that we called Mesh-Gateway Access Node (MG-AN). The MG-ANs have the functionality of the 802.11s MPP (Mesh Portal Point) implemented in the access node (AN) of our mesh infrastructure. So, in order to connect to the Internet cloud, the mesh clients forward, first, their traffic to their own gateway (i.e., MG-AN), for accessing the mesh infrastructure. Then, the MG-AN forwards directly the received traffic from its mesh clients to its own gateway.

V. HYBRID QOS MESH ROUTING

HQMR, our proposed protocol, is used to ensure routing functionalities within the wireless mesh infrastructure of our global wireless mesh architecture. It is a hybrid QoS-based routing protocol composed of two different routing blocks. The first routing sub-protocol Intra-Mesh infrastructure Reactive Routing (IMRR) is designed to forward communications within the infrastructure in a reactive manner, while the second routing block Inter-Mesh infrastructure Proactive Routing (IMPR) is deployed to forward communications to the external networks, particularly to the Internet network. The second routing sub-protocol is a tree-based multipath routing protocol, with the Mesh Gateway as a root of the routing tree.

Moreover, in order to improve the performance of our routing protocol, we adopt the concept of clustering to divide the topology of the infrastructure into a set of groups called clusters, each coordinated by a cluster-head. This division allows the network to minimize effectively the load of the control messages since the exchange would be limited to a cluster domain. It helps also in reducing the size of the routing table at each node and simplifies routes discovery process thanks to the inter-clusters communications approach. Besides, this concept of clustering is considered as the most suitable solution to ensure the network scalability.

In this section, we present the algorithm specified for the clusters elaboration within the wireless mesh infrastructure and we introduce the two routing sub-protocols of HQMR. Before that, we define the mechanism used to provide the needed information about each node's neighbors and we specify the different QoS parameters and their estimation method to guaranty the QoS based routing characteristic of our proposed HQMR protocol.

A. Neighborhood Maintenance

Neighborhood information is very important for our protocol in order to provide the local topology (node's different neighbors), the necessary information for our clustering algorithm and the available QoS toward each neighbor. To maintain this information, every node in the network is required to send out periodically a Hello message (Table I), announcing its existence and its cluster information such as its state in the cluster, its calculated weight parameter used for cluster-head election, its CH's IP address (ID-CH) and its used bandwidth parameter. By receiving the Hello message from the different neighbors, each node updates its Neighbor Table (Table II), which is used to store for each neighbor its IP address (ID), all the needed information for clusters formation (Weight, State, ID-CH) and the available QoS parameters, including the available bandwidth, the delay and the jitter parameters.

TABLE I. HELLO MESSAGE FORMAT

ID	Weight	State	ID-CH	Used Bandwidth

TABLE II. NEIGHBOR TABLE (NT)

ID	Weight	State	ID-CH	QoS Metric

B. QoS Routing Metrics

The purpose of our routing protocol is to find paths, which can satisfy the QoS requirements of real-time flows. The set of QoS requirements includes the bandwidth, the delay and the jitter parameters.

1) Available Bandwidth metric

To estimate the available bandwidth, each node considers the used bandwidth by its flows and the consumption of its neighbors announced in the Hello messages (1).

$$\mathrm{B}(v) = \mathrm{B} - \sum_{v' \in \mathrm{N}(v)} B_{\mathrm{used}}(v') \qquad (1)$$

where $\mathrm{B}(v)$ is the estimated available bandwidth by a node v, B is the total Bandwidth, $\mathrm{B}_{\mathrm{used}}$ is the bandwidth used by a node and $\mathrm{N}(v)$ in the neighborhood of the node v.

Then, the bandwidth parameter of the entire path is determined as the minimum bandwidth estimated at each node toward the destination.

2) *Delay Metric*

This metric estimation is based on measuring the round trip delay time (RTT) [8] of the Hello messages, which represents the time between initiating a Hello message and receiving a response. The delay of a path is the sum of its links delay metric.

3) *Jitter Metric*

The jitter metric defines the delay metric variation. It is estimated by calculating the mean of the differences between the RTT values for a specific period. Besides, the Jitter of a path is calculated by summing the Jitter of each link.

C. *Clusters formation algorithm*

Our clustering algorithm is a variant of the LID-based clustering algorithm [13] combined with the use of the weight concept developed by the Weighted Clustering Algorithm (WCA) [15] for the election of cluster-heads. Thus, a cluster is formed by the node with the lowest weight and all its neighbors. The same procedure is repeated among the remaining nodes, until each node is assigned to a cluster. Inter-clusters connectivity is maintained by defining some Gateway-nodes (Sub-Section 3), named Cluster Gateway (C-Gw) and Distributed Gateway (D-Gw). Moreover, in our adapted algorithm, we have opted for one-hop clusters to reduce the load of control messages within a cluster and to ensure a line of sight between the different cluster-heads and gateway nodes, which is an important characteristic for the deployment of our second routing sub-protocol IMPR (Section E). An example of a clustered wireless mesh infrastructure is illustrated in Fig. 2.

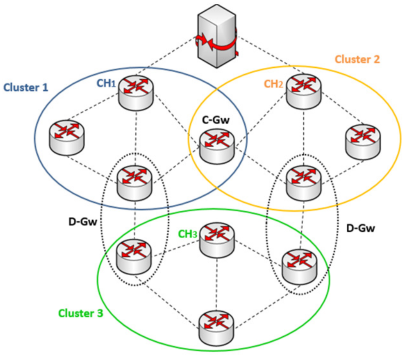

Figure 2. Clustered architecture of the wireless mesh infrastructure

Our clustering algorithm is composed of three main functions, which are presented in the following sub-sections: weight calculation, cluster-head election and clusters elaboration process.

1) *Weight Calculation*

In our algorithm, the weight assigned to each node is based on the WCA algorithm [15]. The latter takes into account the degree (neighbors' number), the transmission power, the mobility and the battery power of each node. It optimizes the degree of each cluster-head by choosing an optimal number M of nodes per cluster (M is a pre-defined threshold). This restriction aims that the cluster-head would be able to support ideally the nodes within its cluster.

However, given the stability of the nodes within our wireless mesh infrastructure, we are only interested in the first two parameters used to calculate the weight of WCA to find the optimal number of nodes within the transmission range and to estimate the transmission power toward the neighbors of a node. In addition, since most of the traffic is oriented to the Mesh Gateway, a third parameter is used in our weight calculation to take into account the power transmission of the node toward the Mesh Gateway. By this way, the cluster-head will be elected among the nearest nodes to the Mesh Gateway. Thus, the weight is calculated according to (2)-(6):

$$Wv = a \cdot \Delta v + b \cdot Dv + c \cdot DPv \tag{2}$$

where a, b and c are the weighing factors so that $a+b+c=1$ and Wv is the weight of a node v.

$$dv = \left|N(v)\right| = \left|\left\{v' \in V, v' \neq v, dist(v,v') \langle tx_{range}\right\}\right| \tag{3}$$

where V is the neighborhood of a node v.

$$\Delta v = \left|dv - M\right| \tag{4}$$

$$Dv = \sum_{v' \in N(v)} dist(v,v') \tag{5}$$

$$DPv = dist(v, MG) \tag{6}$$

Equation (4) represents the degree-difference for a node v to compare its number of neighbors (3) to the optimal number of nodes that a CH may coordinate efficiently. The transmission power toward the neighbors is estimated in (5) by computing the sum of the distances with all its neighbors. Samely, the third parameter namely the transmission power toward the Mesh Gateway is calculated in (6).

2) *Cluster-head Election*

Initially, all the nodes are in the initial state that is the "Undecided" state and with a weight equal to zero. Thanks to the periodic exchange of Hello messages, the Neighbor Table (Table II) will be updated with the last calculated value of weight (W) for each neighbor. Each node waits for a period T_e before starting the selection of the cluster-heads, so that all the nodes have updated their NT (Neighbor Table). After this period, the node with the lowest W among its neighbors changes its state to "CH" and broadcasts a Hello message, as illustrated in Fig. 3.

```
If $W_i$ = min (NT [weight]) then
    $S_i$ = CH
    $ID\text{-}CH_i = ID_i$
    Broadcasts Hello ($ID_i$, $W_i$, $S_i$, $ID\text{-}CH_i$, $B_{used}$)
End If
```

Figure 3. Cluster-head election algorithm for node *i*

3) Clusters elaboration process

The division of the network into a set of clusters is based on the exchange of Hello Messages between each node and its neighbors. Fig. 4 illustrates the algorithm of the clusters elaboration.

```
On receiving a Hello message:

If (Hello [State] = CH) then {
    If ID-CH$_i$ = null then {
        $S_i$ = CM
        ID-CH$_i$ = Hello [ID-CH]
        Update (NT)
        Broadcast Hello (ID$_i$, W$_i$, S$_i$, ID-CH$_i$, B$_{used}$) }
    Else {
        If (Hello [ID] < ID-CH$_i$) then
            G = ID-CH$_i$
            ID-CH$_i$ = Hello [ID]}
        End If
        $S_i$ = C-Gw
        Update (GwT)
        Broadcast GW-D (ID$_i$, W$_i$, C-Gw, ID-CH$_i$, G, null) }
    End If }
Else if (Hello [State] = CM) then {
    If ($S_i$ = CM) then {
        If (ID-CH$_i$ = Hello [ID-CH]) then
            Update (NT)
        End If
        $S_i$ = D-Gw
        Update(GwT)
        Broadcast GW-D (ID$_i$, W$_i$, D-Gw, ID-CH$_i$, Hello [ID-
        CH], Hello[ID])
    }
    Else
        Update (NT)
    End If }
Else {
    If (ID-CH$_i$ = Hello [ID-CH]) then Update (NT)
    Else {
        Update (NT)
        Update (NCHT)
    }
    End If }
End If
```

Figure 4. Clusters elaboration algorithm

According to our algorithm, we distinguish five possible states of a node within a cluster. Besides, it is important to notice that the clustering algorithm is executed on each node of the infrastructure except the Mesh Gateway. The latter has its own state MG as Mesh Gateway. For the rest of nodes, we have the following states:

- **Undecided**: it is the initial state indicating that the node does not yet belong to any cluster.
- **Cluster Member** (CM): it is a node, which belongs already to a cluster. It changes its state from Undecided to CM, once it has received a Hello message from a CH.
- **Cluster Head** (CH): it is the node with the lowest weight and it is responsible of its cluster management.
- **Cluster Gateway** (C-Gw): it is a node in direct vision with two different cluster heads at the same time. It acts as a bridge between the two clusters. In fact, this node exists when two cluster-heads are at two-hops of each other.
- **Distributed Gateway** (D-Gw): it is a CM that has a neighbor belonging to another cluster. D-Gw ensures the communication between two disjoint clusters. This is the case where two cluster-heads are at 3-hops of each other.

These different states with the different necessary transition conditions are decribed in a FSM diagram (Fig. 5).

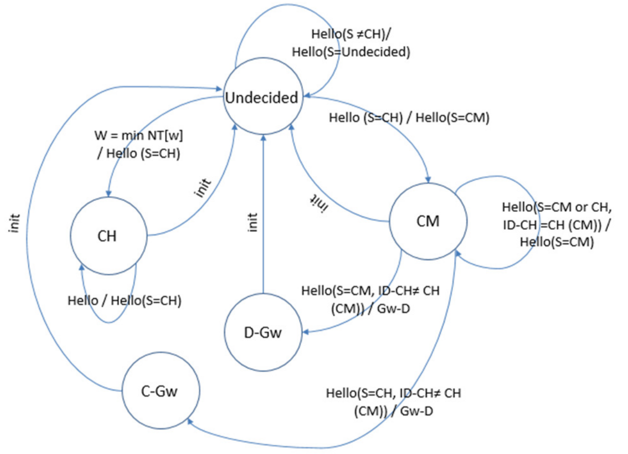

Figure 5. FSM of a node participating in the clustering algorithm

A node becomes a CM node when it receives a Hello message for the first time from a CH node. This node may change its state to a gateway node to ensure interconnection between two clusters. It may become a C-Gw when receiving a Hello message from another cluster-head. It changes its state to a C-GW and updates its Gateway Table (Table III), in which it keeps its type as gateway and the two interconnected cluster-heads. Then, a GW-D (Declare) message (Table V) is sent to its cluster-head and the neighbor cluster-head. By receiving this message, each of the cluster-heads updates its Neighbor CH Table (NCHT) (Table IV), in which it keeps the neighbor cluster-heads and its corresponding gateways. This process is illustrated in the MSC (Message Sequence Chart) diagram [26] in Fig. 6.

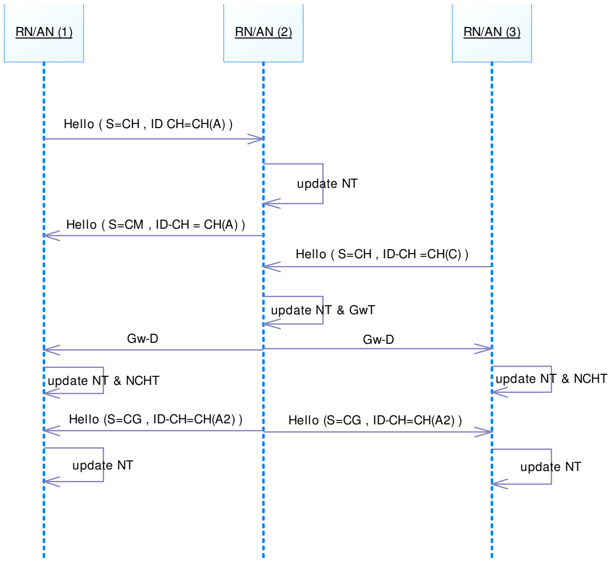

Figure 6. MSC of C-Gw selection scenario

A CM node may also become a D-Gw when receiving a Hello message from a CM belonging to another cluster, as illustrated in Fig. 7.

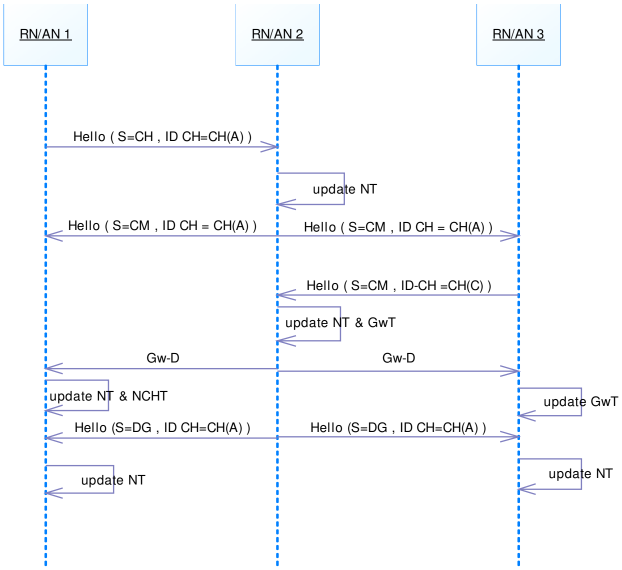

Figure 7. MSC of D-Gw selection scenario

TABLE III. GATEWAY TABLE (GWT)

Type-Gw	ID-CH1	ID-CH2	ID_D-Gw

TABLE IV. NEIGHBOR CLUSTER-HEAD TABLE (NCHT)

Neighbor ID-CH	Gw-ID	Type-Gw

TABLE V. GW-D MESSAGE

ID	Weight	Type-Gw	ID-CH	Neighbor ID-CH

D. *Intra-infrastruture Routing (IMRR)*

Intra-Mesh Infrastructure Reactive Routing (IMRR) is the reactive routing sub-protocol of our proposed HQMR protocol. It is used to find routes in order to forward information between two nodes located within the infrastructure. It ensures QoS based routing for nodes belonging to a same cluster as well for those located in different clusters. Moreover, IMRR offers QoS guarantees by checking the QoS parameters namely bandwidth, delay and jitter at each node during the route discovery process.

Furthermore, the proposed IMRR sub-protocol is an enhancement of AODV routing protocol [27], which takes into account the clustering approach and the QoS verification in route discovery process.

1) *IMRR operation*

Fig. 8 illustrates the algorithm of IMRR operation. A node S starts directly to forward data if the destination D is one of its neighbors, with verified QoS or if a valid route to D exists in its routing table. Otherwise, S launches the route discovery process.

```
When a node S wants to transmit data to a node D:
S verifies its routing table
If a valid route with requested QoS exists then
    Forward (data, D)
Else
    Multicast (RREQ)
End If

On receiving a RREQ message:
If QoS verified then {
    If it is the destination then
        Send (RREP, S)
    Else {
        Update (RREQ)
        Update (RT)
        If it is CH then {
            If D exists in NT then
                Unicast (RREQ, D)
            Else Multicast (RREQ)
            End If }
        Else multicast (RREQ)
        End If
    End If}
Else discard (RREQ)
End If }
```

Figure 8. IMRR operation algorithm

The received RREQ message is either forwarded directly to the destination or forwarded to the multicast group formed by the different CHs, C-Gws, D-Gws and the MG. The use of the multicast group limits the broadcast of the RREQ messages, which helps reducing the load of the network.

According to this algorithm, two nodes from different clusters may communicate with each other only through a route formed by CHs and/or Gws and/or the Mesh Gateway.

An example of a communication between two nodes from different clusters is illustrated by a MSC diagram in Fig. 9.

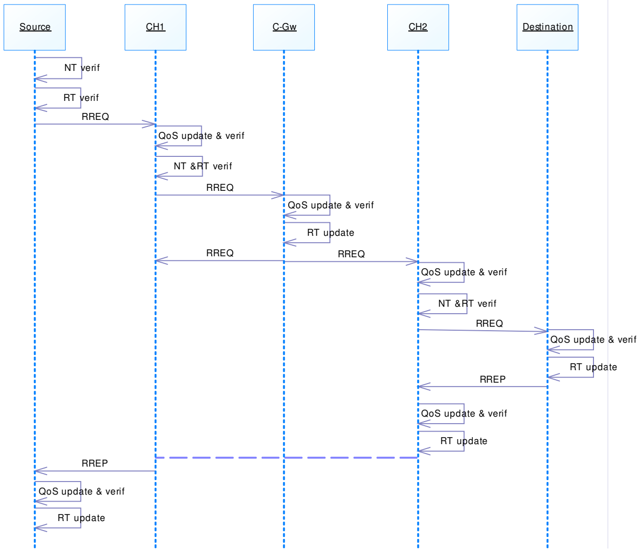

Figure 9. MSC for inter-clusters IMRR routing

2) *Route Discovery Process*

Like AODV protocol, IMRR uses RREQ message for route discovery (Table VI). However, the RREQ message used by our IMRR routing protocol introduces specific QoS fields to enable QoS based routing. Each intermediate node proceeds to a QoS verification before forwarding the request (7).

$$(B_{off}>=B_{req} \text{ or } B=null) \text{ and } (D_{off}>=D_{req} \text{ or } D=null) \text{ and } (J_{off}>=J_{req} \text{ or } J=null) \quad (7)$$

where B is the bandwidth, D is the delay and J is the Jitter.

TABLE VI. RREQ MESSAGE

Src IP address	Dest IP address	Broadcast ID	Path	QoS Metric requested	QoS Metric offered	ID msg

In Fig. 10, we illustrate the processing of a RREQ message at each node. Unlike AODV protocol, only the destination node is able to respond to a RREQ message, so that it would have the entire path's estimated QoS to compare it properly to the requested one. Moreover, the duplicated RREQ messages (Broadcast ID already exists) are not rejected. Instead, we send as much as possible of RREQ messages to the destination to guarantee the discovery of the best path. In order to avoid an infinite loop of a message, each node verifies first if its address already exists in the Path field or not. Then, we introduce a new parameter called "ID msg" to distinguish the duplicate messages at a node. This parameter is updated at each intermediate node for each RREQ message received (duplicated or not). Thus, a node assigns a new "ID msg" for each request and inserts it into the RREQ message. Then, the reverse route is created within the routing table (Table VII), by taking into consideration this parameter, so that it would be used later for the RREP message forward.

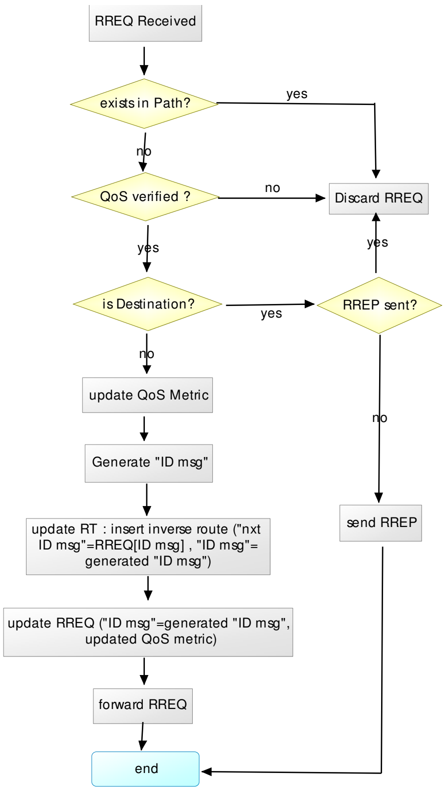

Figure 10. RREQ processing

TABLE VII. IMRR ROUTING TABLE

Dest IP address	Next Hop	Lifetime	QoS Metric offered	ID msg	Nxt ID msg

3) Route Reply Process

In order to establish a route toward the source node, the destination responds with a RREP message (Table VIII) to the first RREQ received verifying the requested QoS parameters and rejects the following RREQ messages. The processing of a RREP message at each intermediate node is illustrated by a flowchart in Fig. 11.

A mesh node determines the next hop thanks to the "ID msg" parameter. It updates then the routing table with the direct route and the "ID msg" with the "Nxt ID msg" of the routing table before forwarding the RREP message.

TABLE VIII. RREP MESSAGE

Src IP address	Dest IP address	Lifetime	QoS Metric requested	QoS Metric offered	ID msg

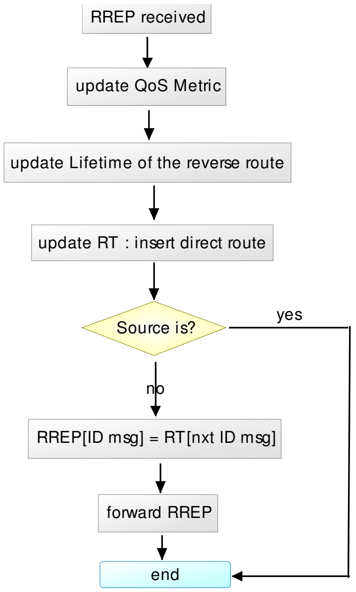

Figure 11. RREP processing

E. *Inter-infrastructure Routing (IMPR)*

Inter-infrastructure Mesh Proactive Routing (IMPR) is the second routing sub-protocol of HQMR, designed to ensure communications toward external networks, especially Internet network. Since most of the traffic goes through the Mesh Gateway to provide Internet services, we opted for a proactive tree based routing protocol, having the Mesh Gateway as a root and the different CHs and C-Gw and/or D-Gw as children. It is important to notice that the different cluster members would not participate in the trees construction process. In Fig. 12, we present the topology of the clustered infrastructure presented in Fig. 2, which we would have if we do not consider the different Cluster members and keep only the nodes that can play the role of children in our routing trees.

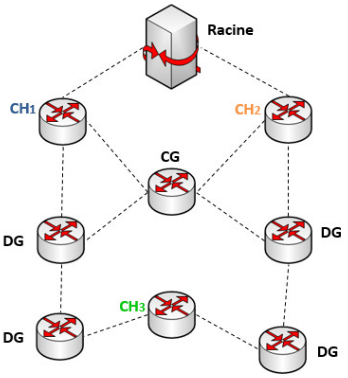

Figure 12. Network topoloy for trees construction

In addition, to provide QoS guarantees for real-time flows, IMPR deploys a multi-path routing concept to define three different routes, partially node-disjoint, between each child and the root. These routes would be used to construct three partially disjoint routing trees within the IEEE 802.16j wireless mesh infrastructure, in such a way that each tree is used to forward a specific type of traffic. To this end, we define for our protocol three service classes, namely interactive real-time applications class, Streaming applications class and Best Effort class. The first class is more sensitive to delay and jitter variations, the second one is more sensitive to jitter variation, and the last class is more exigent in terms of loss ratio. In other words, IMPR allows the construction of three partially disjoint trees with a common root: Real Time, Streaming, and Best Effort Trees.

1) Root Announcement process

The root (i.e., MG) broadcasts a RANN (Route Announcement) message to all its neighbors to announce its presence. This message is considered only by the CHs and the Gws. It is rejected by all the CM nodes. On receiving a RANN message (Table IX), each intermediate node stores the Path parameter in its route cache and updates it next by adding its address. It updates also the QoS Metric and proceeds to the forward of the updated RANN message to its multicast group formed by the CHs, the Gws, and the MG. In order to keep as many routes as possible, duplicated RANN messages are not rejected. Instead, to avoid an infinite loop of a message, each node verifies first if its address already exists in the Path field or not. In fact, each node keeps the entire path received through the RANN message in its route cache in order to be able to verify later the disjunction of two paths.

TABLE IX. RANN MESSAGE

Root IP address	Path	QoS Metric

2) Routing trees construction

Each node waits for a certain time Ts before starting the routing trees construction process, in order to store the maximum of paths. Firstly, using the routes selection algorithm (Fig. 13), each node selects a route for the Real Time Tree. This route is validated as one of the tree branches by an exchange of PREQ and PREP messages with the root. Once the PREP received from the root, each node removes the chosen path from its route cache and starts the construction process of the second routing tree in the same manner. Then, the mechanism is repeated for the third routing tree. In fact, the exchange of PREQ/PREP messages performed for routes validation is used to ensure that each intermediate node of a path is using the same path toward the root, so that each node has no more than a single branch toward the root of a tree.

3) Routes Selection Algorithm

This algorithm is described in Fig. 13. The idea is to select at each node a potential path for each routing tree, satisfying the requirements of the defined service classes. For the first path corresponding to the Real Time Tree, we choose the best in terms of delay and jitter with satisfying bandwidth metric. The second one should be partially disjoint from the first one to reduce congestion issues, with good values of the jitter QoS parameter. Lastly, from the remaining paths, we select the best in terms of disjunction over the other paths.

Some nodes may not be able to select three different paths. Thus, for the case where a node has only selected two paths, the first one would be used to forward the highest priority traffic, while the second one would be shared between the two other service classes. If only one path is present at a node, we adopt the default QoS mechanism of IEEE 802.16j to share it between the three service classes.

```
P ← set of stored paths ; Disj ← number of common nodes between paths
HC: Hop Count ; $w_i$ : QoS parameters' weight ; L: weight of a path

If $tree_i = 1$ then
  $A = \{P\}_{(D<Dmax \text{ and } J<Jmax)}$
  If $A \neq \Phi$ then
    $P_1 = \min_{HC} \{\max_{Bw} A\}$
  Else
    $B = \{P\}_{(D<Dmax)}$
    If $B \neq \Phi$ then
      Calculate $L=w_1*rank_{desc}Bw + w_2*rank_{asc}J$ for each path in B
      $P_1 = \min_L B$
    Else
      Calculate $L=w_1*rank_{desc}Bw+w_2*rank_{asc}D+w_3*rank_{asc}J$ for each Path in P
      $P_1 = \min_L P\}$
    End If
  End If
End If
If $tree_i=2$ then
  $P = P\backslash\{P_1\}$
  $A = \{P\}_{(J<Jmax)}$
  If $A \neq \Phi$ then
    Calculate $L=a*rank_{desc} Bw+b*rank_{asc} Disj+c*rank_{asc}J$ for each Path in A
    $P_2 = \min_L A$
  Else {
    $L = w_1*rank_{desc} Bw + w_2*rank_{asc} J + w_3*rank_{asc} Disj$ for each Path in P
    $P_2 = \min_L P$
  End If
End If
If $tree_i=3$ then
    $P = P\backslash\{P_2\}$ ; $P_3 = \min_{HC} \{\min_{Disj} P\}$
End If
```

Figure 13. IMPR routes selection algorithm

4) Path Request Process

By executing the route selection algorithm, a node selects a path for its i^{th} routing tree and sends a PREQ message (Table X). Each intermediate node compares its chosen path for its ith routing tree to the path carried by the PREQ message. If the next hop in the two paths is different, the node either modifies its entire path or updates the path in the PREQ message, as presented in the Flowchart in Fig. 14. Then, the intermediate node updates its routing table (Table XI) with both the direct route (toward the root) and the reverse route (toward the source) and forwards the PREQ message to the next hop.

TABLE X. PREQ MESSAGE

Src IP address	Dest IP address	Path	ID-Path	Level[a]

[a] Level : the level of a node in the Real Time tree

TABLE XI. IMPR ROUTING TABLE

Dest IP address	Next Hop	ID-Path

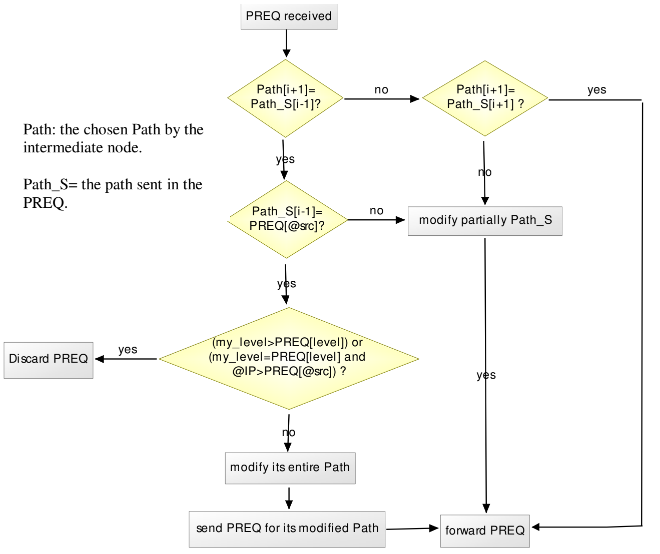

Figure 14. Flowchart of PREQ process

5) *Path Replay Process*

On receiving the PREQ message, the root updates its routing table and sends a PREP (Table XII) message to its child.

TABLE XII. PREP MESSAGE

Dest IP address	Path	ID-Path

Each intermediate node adds its address to the Path parameter of the PREP message and forwards it to the destination. Once the destination receives the PREP message, it updates its routing table and its chosen path for the routing tree if it is different from the Path parameter in the PREP message. Then, it removes it from its route cache to begin the selection of a route for the next tree.

VI. HQMR USAGE SCENARIOS

In this section, we present two different usage scenarios of our HQMR protocol, describing how a path is selected to reach a destination within or outside the mesh infrastructure.

A. *Intra-infrastructure Routing Usage Scenario*

This scenario describes how to determine a QoS verified path between two nodes from different clusters for a VoIP application between two mesh clients of our architecture. To this end, the reactive routing bloc, named IMRR would be used and a RREQ message is generated for route discovery process. In Fig. 15, we illustrate the RREQ process through each intermediate node by comparing the offered QoS to the requested one (B_{req}=56Kb/s, D_{req}=150ms and J_{req}=20ms).

The first RREQ received by D (<2, 155, 19>) does not satisfy the requested delay parameter. Thus, this message is discarded and D waits for another RREQ messages. Since the second message received (RREQ2) verifies the different QoS parameters (<2, 145, 13>), a RREP message is unicasted to the source node.

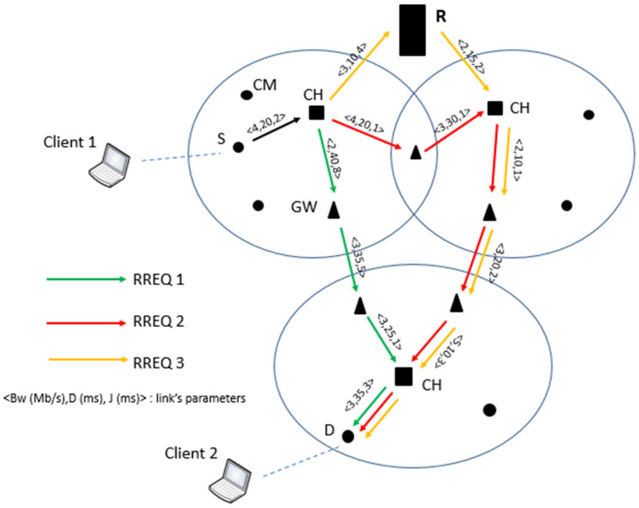

Figure 15. Intra-infrastructure usage scenario

Then, regarding the third RREQ message received, it would be discarded since a RREP message has been already sent back. By this way, the route discovered by RREQ2 would be used to forward the traffic of the VoIP application between the two mesh clients.

B. *Inter-infrastructure Routing Usage scenario*

For communications with the Internet network, the proactive routing protocol IMPR of HQMR protocol is used. In this scenario, we describe how to forward a VoD (Video on Demand) application traffic from a streaming video server in the Internet. To this end, three QoS based routing trees are constructed. Fig. 16 shows the topology used for this scenario. Then, we illustrate in Fig. 17 the three routing trees built over this topology to forward traffic to the Internet networks.

For the first routing tree, by executing the route selection algorithm (D_{max}=150 ms, J_{max}=20 ms), we chose the paths with satisfying delay and Jitter parameters. To better explain the construction process of this tree at each node, we describe for example the case of the node B. We have four paths towards the root satisfying the delay and jitter parameters: B-A-R: <4,30,11> ; B-C-R: <3,70,6> ; B-D-A-R: <3,95,16> ; B-E-C-R: <2,70,9>. Then, the path with the highest Bandwidth is selected: B-A-R. This process is repeated at each node of the topology and a PREQ message is sent for route validation. Once a node receives a PREP message for the first route, it starts the selection of a path for the second routing tree.

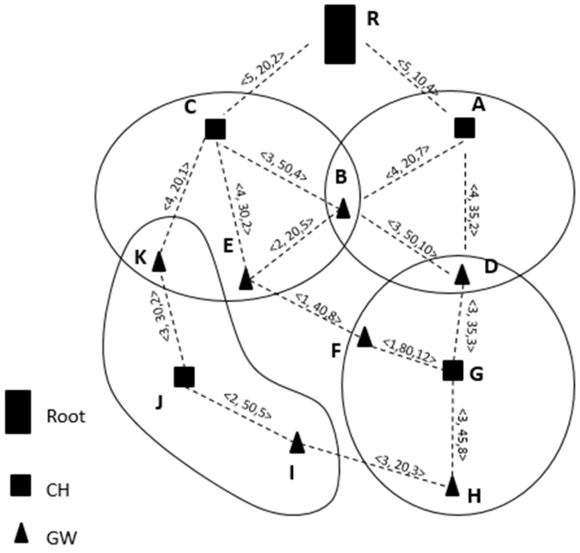

Figure 16. Example of clustered wireless mesh infrastructure

Similarly, the path at each node for the second routing tree is selected according to the routes selection algorithm. For example, at the node E, we have two paths verifying the jitter parameter: E-B-C-R: <2, 90, 11>; E-B-A-R: <2, 50, 16>. The second path is the one selected by E since it offers better disjunction with the path selected for the first tree. However, the PREQ of this message would be changed at the node B. In fact, the nodes B and E have the same level parameter but node B has a greater IP address than E. Thus, the path in the PREQ sent by the node E would be changed (to E-B-C-R) to correspond to the route chosen by the node B: B-C-R.

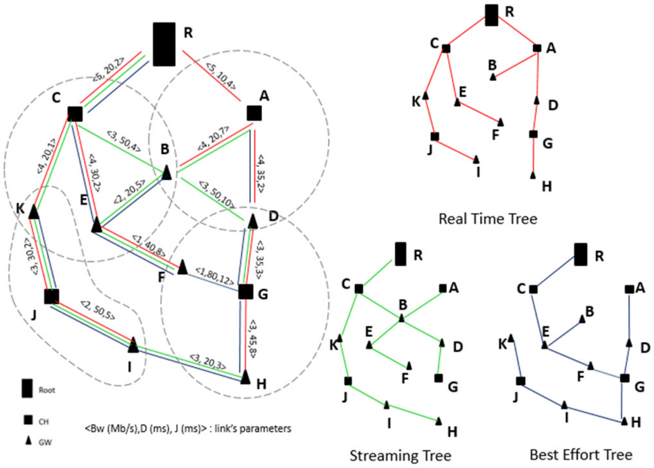

Figure 17. The QoS based routing Trees

By receiving the PREP message for the selected route, each node starts the selection of its third route that is the most disjoint route to the two first selected paths with a minimum of hops. For example, the node I according to these conditions chooses the path I-H-G-D-A-R. However, its PREQ at the node G would be changed (see the flowchart in Fig. 14). The selected path by the node I would be modified partially (I-H-G-F-E-C-R) to correspond to the one selected by the node G. Then, the set of paths selected at each node forms the routing tree for the third service class.

Regarding our usage scenario, and in order to forward the VoD traffic, the second routing tree would be used since this application is considered as an application of the Streaming service class.

VII. PERFORMANCE EVALUATION OF IMRR

To evaluate the performance of our routing protocol, we have developed the source code of HQMR protocol using the network simulator ns-3. Then, we have conducted some simulation scenarios to compare our IMRR routing sub-protocol and the AODV protocol in a wireless mesh environment.

In this section, we define briefly the used network simulator namely ns-3. Then, we present the different performance parameters that we have evaluated as well as the simulation environment.

A. *Network Simulator Ns-3*

Ns-3 [28] is a discrete-event network simulator, developed within the ns-3 project, started in 2006 to eventually replace the aging ns-2 simulator. It is a free open software, which can be used for research to contribute and share the developed codes. The ns-3 software is built on C++, containing a set of network simulation modules implemented as C++ objects. Then, the simulation scripts may be written in C++ or Python.

The existing library of modules allows simulating popular wireless networks (e.g., Wi-Fi, WiMAX, LTE) in a simple way. Moreover, thanks to the availability of the source code of existing modules, it is possible to modify the operation of any module from the library. It is also possible to create new modules, implementing algorithms or protocols not present in the existing library.

B. *Performance metrics*

In our simulations, we have measured the following key performance parameters to evaluate the effectiveness of our IMRR routing sub-protocol.

- Routing Overhead: the total number of control messages generated during the simulation for route discovery process. For messages sent over multiple hops, each transmission (each hop) counts as one generated control message.
- Route Discovery Convergence: the necessary time to establish the route between the source node and the destination one. It is the time between the received RREP message and the RREQ message being sent by the source.

- Average Throughput: It is the mean of the number of packets successfully transmitted to their final destination per unit time.
- Average End-to-End Delay: This is the overall average delay required by a packet to travel from source node to its destination node. It includes all possible delays caused by queuing delay, retransmission delays and propagation delay.
- Average Jitter: It is the mean of the difference between the end-to-end delay values.

C. *Simulation Environnement*

The simulation environment consists up to 25 stationary mesh nodes arranged in a grid topology. Simulation time is 50 seconds. Each scenario is simulated five times and an average value is taken for the performance analysis. Table XIII shows the used simulation parameters.

TABLE XIII. SIMULATION PARAMETERS

Simulation parameters	**Value**
Routing Protocols	IMRR & AODV
Simulation Time	50s
Nodes 'Number	6 to 25 nodes
Mobility Model	GridPositionAllocator / static
Traffic Model	CBR (UDP) / VoIP (UDP)
Packet Size	512 bytes / 160 bytes
DataRate	512kbps / 64 kbps

To analyze the different performance metrics, two different traffic models are used in the elaborated scenarios. The first one is a generic CBR traffic used in scenario 1 (see Section VIII.A) and the second one simulates a VoIP traffic used in scenario 2 (see Section VIII.B).

VIII. SIMULATION RESULTS AND ANALYSIS

A. *Scenario 1*

To evaluate our protocol performance in terms of routing overhead and route discovery convergence, we perform different simulations, by varying the number of nodes within our mesh infrastructure, while considering a constant-bitrate (CBR) traffic. This traffic is modeled with 512-byte data packets and a data rate of 512kbps.

In fact, the routing overhead for route discovery process includes, for both AODV protocol and IMRR sub-protocol, the Hello messages, the RREQ messages and the RREP messages. Figure 18 shows the variation of the number of RREQ messages according to the number of nodes in the network. We notice that the amount of RREQ messages of HQMR protocol increases with the number of nodes while AODV protocol presents a low variation. This is explained by the fact that AODV protocol rejects each duplicated RREQ message at each node. By this way, less RREQ messages are forwarded in the network. However, in order to ensure the best route discovery in terms of QoS guarantee, HQMR does forward some duplicated received RREQ messages to the destination (up to three at each node). On the other hand, AODV presents a higher amount of RREP and Hello messages, as illustrated in Fig. 19 and Fig. 20. Indeed, we observe a major difference between the two protocols in terms of RREP messages since AODV protocol allows that an intermediate node replies to a RREQ message. However, with the HQMR protocol, only the destination is allowed to reply to a RREQ message after verifying the offered QoS parameters of the entire received route. Moreover, each time a node sends a RREQ message, sending a Hello message is deferred. Thus, the amount of forwarded Hello messages is inversely proportional to the amount of RREQ messages, which explains the results illustrated in Fig. 18 and Fig. 20.

Since the HQMR protocol forwards less RREP messages than the AODV protocol for all network sizes and since the amount of Hello messages is offset by the RREQ messages and inversely, we can conclude, from these results, that the IMRR sub-protocol has a better global routing overhead.

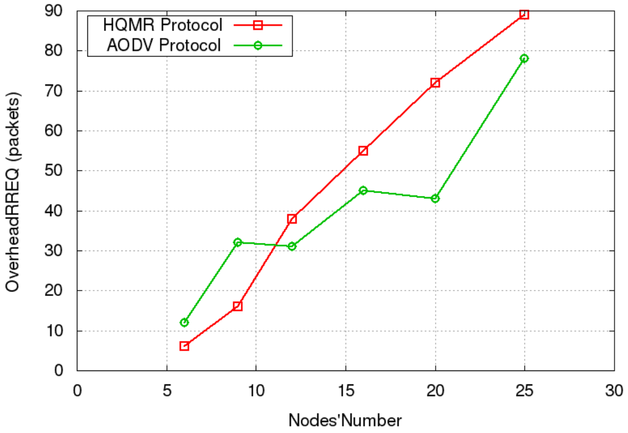

Figure 18. RREQ Messages Overhead

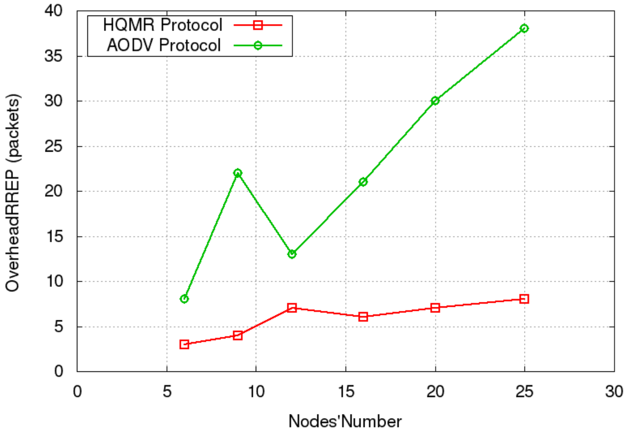

Figure 19. RREP Messages Overhead

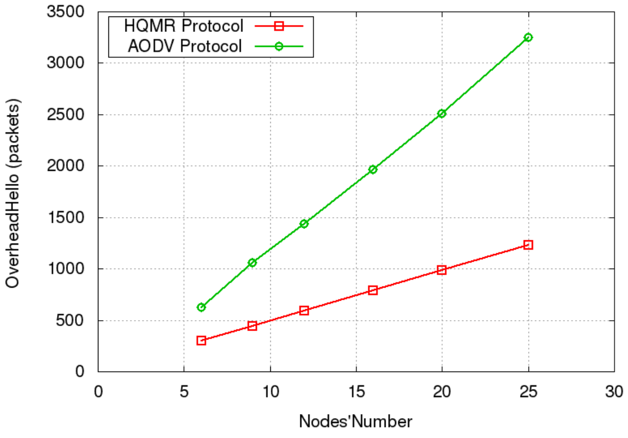

Figure 20. Hello Messages Overhead

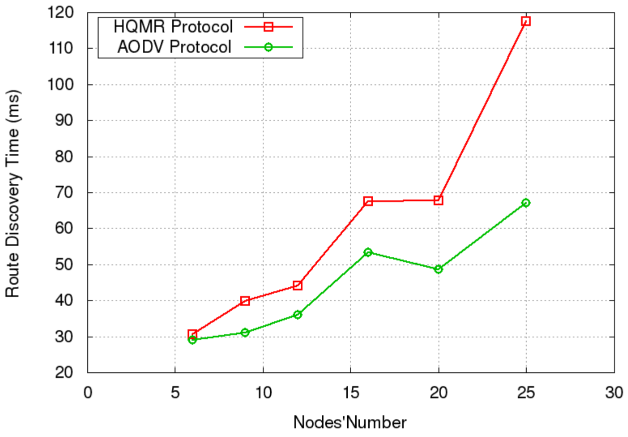

Figure 21. Route Discovery Convergence

Figure 21 illustrates the variation of the route discovery time performance parameter according to the size of the mesh topology. For both protocols, the route discovery convergence time increases with the number of nodes since the discovered route may be longer in terms of hops. Although IMRR sub-protocol has higher values than AODV, we perceive a close variation with the AODV values. Actually, AODV uses the minimum hops number as a metric. That is why, AODV spends less time in discovering routes. However, HQMR protocol accepts longer routes as it offers better QoS parameters to ensure a better routing for real-time and streaming applications.

B. Scenario 2

To underline the effectiveness of our routing protocol in route discovery enabling to correctly forward real-time applications, we evaluate the corresponding performances of such applications in terms of average throughput, average end-to-end delay and average jitter parameters within a mesh topology, since this type of application, i.e., interactive applications, is very sensitive to delay and jitter variation.

To simulate a voice conversation, we used a traffic pattern corresponding to the G711 encoder, which produces 50 packets per second with 160 bytes of payload each. Then, we have introduced a noise over some links to simulate network perturbation. The simulations are conducted to compare the IMRR sub-protocol and the AODV protocol by varying the network size.

The comparative results of throughput, end-to-end delay and jitter QoS parameters are shown in Figs. 22, 23, and 24, respectively. We observe that HQMR offers better values of the average throughput than the AODV protocol for the different mesh network sizes. Besides, we notice a considerable difference concerning the variation of the delay and jitter parameters. The corresponding values while using AODV are more than twice the QoS values while using HQMR.

Thus, the HQMR protocol offers a better route than AODV, especially in terms of delay and jitter, to forward VoIP traffic. Actually, the AODV protocol does not take into consideration the state and the QoS offered by the different links within the mesh topology. It is based only on the number of hops for its route selection. On the other hand, HQMR process a QoS verification during the route discovery process to determine the route satisfying the requested QoS parameters depending on the type of application to forward.

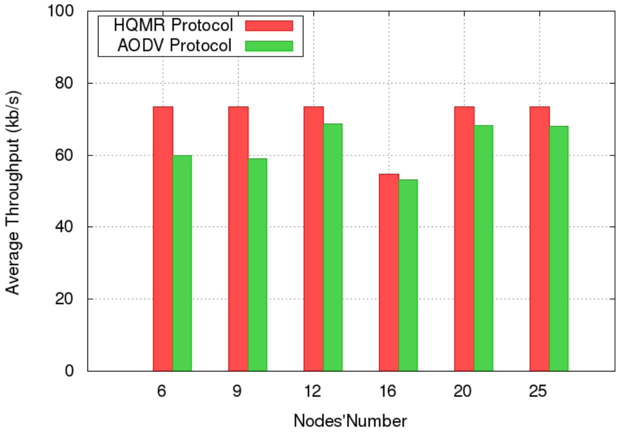

Figure 22. Average Throughput of a VoIP Application

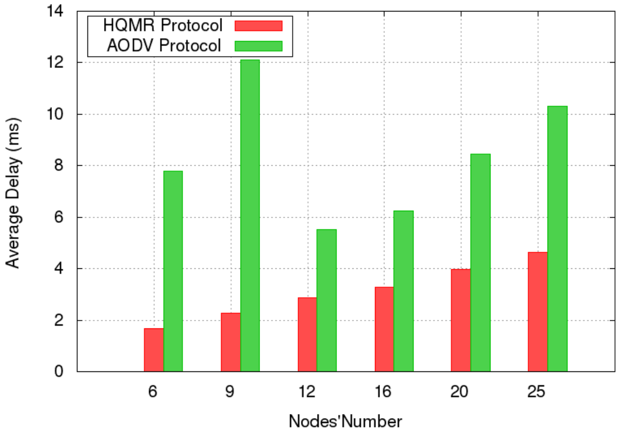

Figure 23. Delay Evaluation of a VoIP Application

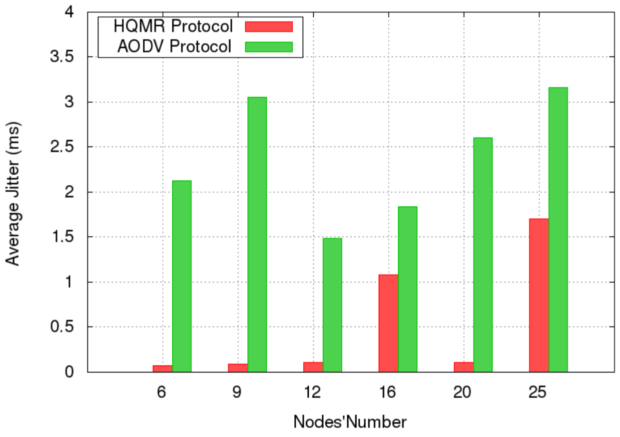

Figure 24. Jitter Evaluation of a VoIP Application

IX. CONCLUSION

In this paper, we presented our proposed hybrid wireless mesh architecture composed of two different domains: an IEEE 802.16j-based infrastructure domain and several IEEE 802.11s based client domains. Then, we have specified the HQMR protocol for ensuring routing functionalities within the wireless mesh infrastructure of our global architecture. It is a hybrid QoS based routing protocol formed by a reactive routing sub-protocol for a clustered infrastructure and a proactive multipath tree based routing sub-protocol for communications toward Internet network. Two usage scenarios are presented to show the importance of HQMR in order to provide real time and streaming applications with QoS guarantee in wireless mesh networks. Then, we presented the different simulations scenarios conducted to evaluate the performance of our IMRR routing sub-protocol in terms of routing overhead and route discovery convergence time, as well as the performance of a real-time interactive application (VoIP) in terms of average throughput, average end-to-end delay and average jitter while using our routing protocol in a mesh topology.

ACKNOWLEDGMENT

This work has been funded by the Regional Council of Burgundy, France.

REFERENCES

[1] H. Bargaoui, N. Mbarek, O. Togni, and M. Frikha, "Hybrid QoS Based Routing for IEEE 802.16j Mesh Infrastructure," AICT 2014, The Tenth Advanced International Conference on Telecommunications, 2014, pp. 110-118.

[2] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," *Comput. Netw.*, vol. 47, n^o 4, pp. 445-487, March 2005.

[3] I. F. Akyildiz and X. Wang, "A survey on wireless mesh networks," *IEEE Commun. Mag.*, vol. 43, n^o 9, pp. S23-S30, 2005.

[4] "IEEE Standard for Information Technology–Telecommunications and information exchange between systems–Local and metropolitan area networks–Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 10: Mesh Networking," *IEEE Std 80211s-2011 Amend. IEEE Std 80211-2007 Amend. IEEE 80211k-2008 IEEE 80211r-2008 IEEE 80211y-2008 IEEE 80211w-2009 IEEE 80211n-2009 IEEE 80211p-2010 IEEE 80211z-2010 IEEE 80211v-2011 IEEE 80211u-2011*, pp. 1-372, 2011.

[5] "IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Broadband Wireless Access Systems Amendment 1: Multihop Relay Specification," *IEEE Std 80216j-2009 Amend. IEEE Std 80216-2009*, pp. 1-290, 2009.

[6] "IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands and Corrigendum 1," *IEEE Std 80216e-2005 IEEE Std 80216-2004Cor 1-2005 Amend. Corrigendum IEEE Std 80216-2004*, pp. 0_1-822, 2006.

[7] Q. Xue and A. Ganz, "QoS Routing for Mesh-Based Wireless LANs," *Int. J. Wirel. Inf. Netw.*, vol. 9, n^o 3, pp. 179-190, July 2002.

[8] R. Draves, J. Padhye, and B. Zill, "Comparison of routing metrics for static multi-hop wireless networks," New York, NY, USA, 2004, pp. 133–144.

[9] V. Kone, S. Das, B. Y. Zhao, and H. Zheng, "QUORUM: quality of service in wireless mesh networks," *Mob Netw Appl*, vol. 12, n^o 5, pp. 358–369, December 2007.

[10] L. Liu, L. Zhu, L. Lin, and Q. Wu, "Improvement of AODV Routing Protocol with QoS Support in Wireless Mesh Networks," *Phys. Procedia*, vol. 25, pp. 1133-1140, 2012.

[11] Y. Zhang, Y. Wei, M. Song, and J. Song, "R-AODV: Rate aware routing protocol for WiFi mesh networks," 2006 IET International Conference on Wireless, Mobile and Multimedia Networks, 2006, pp. 1-4.

[12] X. Cheng, P. Mohapatra, S. Lee, and S. Banerjee, "MARIA: Interference-Aware Admission Control and QoS Routing in Wireless Mesh Networks".

[13] C. R. Lin and M. Gerla, "Adaptive clustering for mobile wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 15, n^o 7, pp. 1265-1275, 1997.

[14] P. Krishna, N. H. Vaidya, M. Chatterjee, and D. K. Pradhan, "A cluster-based approach for routing in dynamic networks," *SIGCOMM Comput Commun Rev*, vol. 27, n^o 2, pp. 49–64, April 1997.

[15] M. Chatterjee, S. K. Das, and D. Turgut, "WCA: A Weighted Clustering Algorithm for Mobile Ad Hoc Networks," Clust. Comput., vol. 5, no 2, pp. 193-204, April 2002.

[16] Z. Haas, M. Pearlman, and P. Samar, "The Zone Routing Protocol (ZRP) for Ad Hoc Networks". http://tools.ietf.org/html/draft-ietf-manet-zone-zrp-04. Consulted : 2015-05-18.

[17] M. Singh, S. G. Lee, T. W. Kit, and L. J. Huy, "Cluster-based routing scheme for Wireless Mesh Networks," 13th International Conference on Advanced Communication Technology (ICACT), 2011, pp. 335-338.

[18] D. Kaushal, A. G. Niteshkumar, K. B. Prasann, and V. Agarwal, "Hierarchical Cluster Based Routing for Wireless Mesh Networks Using Group Head," International Conference on Computing Sciences (ICCS), 2012, pp. 163-167.

[19] Jack Tsai and Tim Moors, "A Review of Multipath Routing Protocols:From Wireless Ad Hoc to Mesh Networks," ACoRN Early Career Res. Workshop Wirel. Multihop Netw., 2006.

[20] R. Krishnan and J. A. Silvester, "Choice of allocation granularity in multipath source routing schemes," in Networking: Foundation for the Future, IEEE INFOCOM '93. Proceedings.Twelfth Annual Joint Conference of the IEEE Computer and Communications Societies, 1993, pp. 322-329, vol.1.

[21] N. S. Nandiraju, D. S. Nandiraju, and D. P. Agrawal, "Multipath Routing in Wireless Mesh Networks," IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS), 2006, pp. 741-746.

[22] M. R. Pearlman, Z. J. Haas, P. Sholander, and S. S. Tabrizi, "On the impact of alternate path routing for load balancing in mobile ad hoc networks," First Annual Workshop on Mobile and Ad Hoc Networking and Computing, MobiHOC, 2000, pp. 3-10.

[23] X. Hu and M. J. Lee, "An efficient multipath structure for concurrent data transport in wireless mesh networks," Comput. Commun., vol. 30, no 17, pp. 3358-3367, November 2007.

[24] M. S. Siddiqui, S. O. Amin, J. H. Kim, and C. Hong, "MHRP: A Secure Multi-Path Hybrid Routing Protocol for Wireless Mesh Network," IEEE Military Communications Conference, MILCOM, 2007, pp. 1-7.

[25] "IEEE Draft Standard for Information Technology-Telecommunications and information exchange between systems-Local and Metropolitan networks-Amendment 8: IEEE 802.11 Wireless Network Management," IEEE P80211vD160, pp. 1-428, November 2010.

[26] D. Harel and P. S. Thiagarajan, "Message Sequence Charts," in UML for Real, L. Lavagno, G. Martin, et B. Selic, Éd. Springer US, 2003, pp. 77-105.

[27] "Ad hoc On-Demand Distance Vector Routing (AODV)". http://www.ietf.org/rfc/rfc3561.txt. Consulted : 2015-05-18.

[28] "ns-3". http://www.nsnam.org/. Consulted : 2015-05-18.