

# A solution for QoS support in wireless ad hoc networks

Leila Boukhalfa, Pascale Minet, Serge Midonnet

► **To cite this version:**

Leila Boukhalfa, Pascale Minet, Serge Midonnet. A solution for QoS support in wireless ad hoc networks. Huitièmes Journées Doctorales en Informatique et Réseaux (JDIR'07), Jan 2007, Marne-la-Vallée, France. pp.35-44. hal-00620270

**HAL Id: hal-00620270**

**<https://hal-upec-upem.archives-ouvertes.fr/hal-00620270>**

Submitted on 6 Dec 2014

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A Solution for QoS Support in Wireless ad hoc Networks

L. Boukhalfa<sup>1,3</sup>, P. Minet<sup>2</sup>, S. Midonnet<sup>1,3</sup>

<sup>1</sup> ESIGETEL  
1 Rue du Port de Valvins  
77210 Avon, France

[leila.boukhalfa, serge.midonnet}@esigetel.fr](mailto:{leila.boukhalfa, serge.midonnet}@esigetel.fr)

<sup>2</sup> INRIA, Rocquencourt,  
78153 Le Chesnay Cedex, France  
[pascale.minet@inria.fr](mailto:pascale.minet@inria.fr)

<sup>3</sup> IGM-Université de Marne la vallée, 77454  
Marne la Vallée, France  
[serge.midonnet@univ-mlv.fr](mailto:serge.midonnet@univ-mlv.fr)

**Abstract**— More and more applications supported by MANETs require Quality of Service (QoS). Much research has been done to date on QoS in ad-hoc networks. In this paper, we present our solution to provide QoS support through different QoS components. We show that the scheduling component does not suffice to satisfy QoS flow requirements because of radio interferences. Interferences could decrease the application throughputs. This can be a real problem for applications that need bandwidth guarantees. To offer guarantees to QoS flows, we propose a solution made up of five QoS components.

## I. INTRODUCTION

A Mobile Ad-hoc NETWORK (MANET) is an autonomous system of mobile nodes connected by wireless links. It is self-organizing, rapidly deployable and requires no fixed infrastructure. Ad-hoc networks have known a great success and now, they are opening up to civilian applications having requirements of Quality of Service (QoS)[1]. Hence, achieving QoS[3] in MANET corresponds to a real need. The QoS, requested from the network, could be defined in terms of one or a set of parameters such as delay, bandwidth, packet loss, delay and jitter. MANET networks are faced with specific constraints: a) the limited bandwidth because of the reduced available radio resources, b) the highly dynamic topology because of versatile radio propagation and the node's mobility, c) the power constraints because network nodes rely on battery power for energy. These MANET specificities make difficult to achieve QoS in these networks.

The goal of this paper is to present a QoS support taking into account the interferences to provide the bandwidth requested by QoS flows. The remainder of this paper is organized as follows. In section II, we present some elements influencing QoS. Section III shows how to support QoS in ad-hoc networks through different mechanisms. We also show that the WCBQ scheduler does not take into account radio interferences. We then describe our solution for QoS support in section IV. Finally we conclude this paper and present some future perspectives.

## II. FACTORS INFLUENCING QoS

Several factors can have an impact on the QoS perceived by the user. Among these factors, we emphasize on the scheduling, the routing and the interferences.

### A. WCBQ Scheduler

In a network, packet scheduling policy refers to the decision process used to select the next packet that will be transmitted. At present, many schedulers are used in wired networks such as First In First Out (FIFO), Stochastic Fair Queueing (SFQ), Fair Queueing (FQ), and CBQ. Whereas in wireless networks, only FIFO and PriQueue schedulers are used.

The scheduling policy adopted in our solution is inspired from the one used in wired networks. We recall that our aim is the QoS support in ad-hoc networks in order to differentiate services between different traffic classes. One solution is to provide a minimum part of the requested bandwidth to different traffic classes. This means that the medium capacity must be shared between traffic classes. We are then interested in the CBQ scheduler [4] (Class Based Queueing) and we have extended it to the wireless environment. CBQ aims at carrying out two goals. The first one is that each class should be able to receive roughly its allocated bandwidth. The secondary one is that when some class is not using its allocated bandwidth, the distribution of the excess bandwidth among the other classes should not be arbitrary, but should be done according to their relative allocations. Hence, CBQ leads to a good resource utilization. To fulfill these two goals, CBQ requires three modules:

- *Classifier*: it inserts packets ready to be sent by the node in the appropriate class queue.
- *Estimator*: it estimates the bandwidth used by each class in the appropriate time interval. This information is used to determine whether or not each class has received its allocated bandwidth.
- *Selector*: using the information from the estimator, it has to decide which class queue is allowed to send a packet. According to [4, 5], a selector should implement two mechanisms which are the general scheduler and the link

sharing scheduler. The general scheduler is to be used to schedule the class queues if the allocated bandwidth for each class can meet the requirement. Otherwise, the link-sharing scheduler is used to adjust the transmission rates.

WCBQ associates with each class a queue, a priority and an allocated bandwidth (see Fig.1). DATA\_Queueues are dedicated to receive data packets. As we distinguish two classes of flows (see section IV.1), we assign distinct data queues to each flow class. CTRL\_Queue is dedicated to receive control packets (e.g. routing packets). This queue has the highest priority, thus, it is served before DATA\_Queueues.

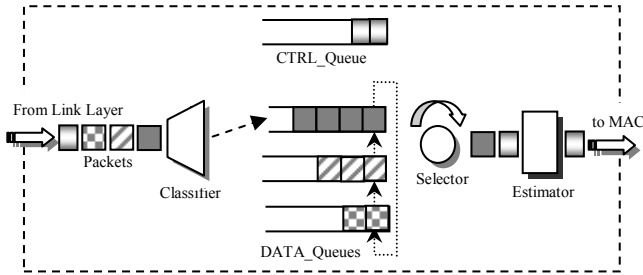


Fig.1. WCBQ mechanism

In [2], we have shown by means of simulations that WCBQ provides the following properties:

*P1:* it shares the node bandwidth between flows present on the node proportionally to their weights.

*P2:* it minimizes the standard deviation of the average bandwidth except for forwarded flows with low throughput.

*P3:* it minimizes the end-to-end delay except for forwarded flows with low throughput.

*P4:* it minimizes the standard deviation of the end-to-end delay for all flows.

### B. Optimized Link State Routing Protocol

OLSR [6] is an optimization of the wired link state routing protocol OSPF [7] for mobile ad-hoc networks. Its innovation lies on the fact that it uses the MultiPoint Relay (MPR) technique. The MPRs of a node corresponds to a subset of its one hop neighbors that allows to reach (in terms of radio range) all two-hop nodes (see Fig. 2). The MPRs technique allows to reduce the control packet size (each node declares only the links with its one hop neighbors that selected it as MPR), and reduces the number of retransmissions when flooding control messages in the network: only the MPRs of the sender forward its packets.

OLSR considers two types of control messages which are neighbor messages, denoted “Hello”, and Topology Control messages, denoted “TC”. The first ones are received by all one hop neighbors, but they are not forwarded to further nodes. The second ones are flooded in the entire network via the MPR nodes. Periodically, each node broadcasts a Hello message containing the information about its neighbors and their link status. This allows each node to: (i) learn its

neighbors at one and two hops and hence construct its neighbor table; (ii) select its MPRs among its one hop neighbors to cover all its two-hop neighbors. In order to construct a topology table, each MPR node periodically sends a TC message containing the list of neighbors that have selected the source of the TC as a multipoint relay.

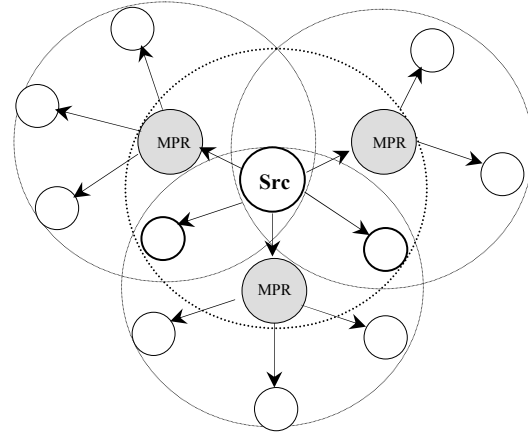


Fig.2. Multipoint relays of node Src

TC messages are forwarded in the entire network. Only the MPR nodes of the sender retransmit the received TC. This avoids useless retransmissions and hence optimizes flooding in the network. Each node of the network maintains a topology table, in which it records the information about the network topology obtained from TC messages.

The routing table is built from the information contained in the neighbor and topology tables, using the Dijkstra algorithm. Therefore, if any of these tables is changed, the routing table is recalculated to update the route to each known destination in the network. Thus, OLSR provides optimal routes in terms of number of hops, which are immediately available when needed.

### C. Radio Interferences

In ad hoc networks, the radio medium being shared, each packet is physically received by all nodes in the transmission range of the sender, whereas nodes in the interference area only detect a busy medium. In IEEE802.11 networks, the medium access is done by a CSMA/CA (Carrier Sense Multiple Access/Collision Avoidance) protocol, like for instance the 802.11b. Collisions lead to a decrease in the medium useful throughput [9].

Let us consider a scenario of 8 nodes and one flow  $f_1$ . The flow  $f_1$  requests a bandwidth of 300kb/s.  $f_1$  is generated by node N0 toward node N9 (fig.3). To illustrate the interference phenomenon, we measure the consumed bandwidth at the MAC level on route node N6 and on node N2 located in the interference area of route node N0.

## III. QoS SUPPORT FOR WIRELESS AD HOC NETWORKS

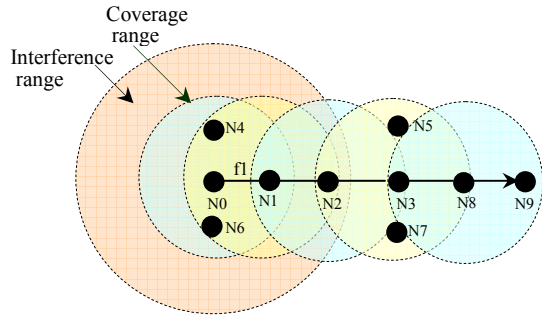


Fig.3. Interference phenomenon

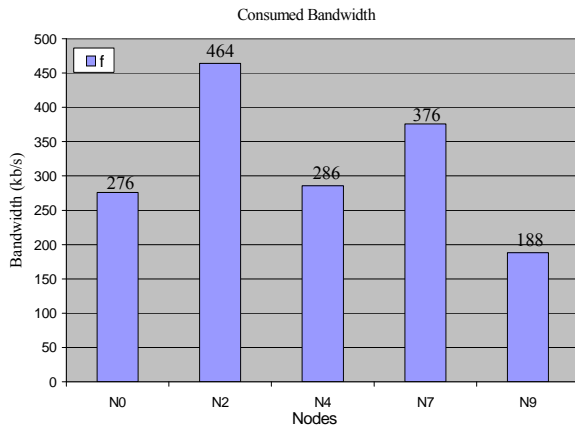


Fig.4. Measured consumed Bandwidth

We note that (see Fig.4) flow  $f1$  has consumed 464kb/s on N2. It represents nearly five times the bandwidth requested by  $f1$ . Indeed, node N2 is disrupted by any packet of flow  $f1$ , once when N0 transmits, because N2 is in the interference area of N0, a second time when N1 transmits because N2 is in the transmission range of N1, a third time when N3 transmits because N2 is in its transmission range, a fourth time when N8 transmits because N2 is in its interference area, and a fifth time when the node itself transmits. For node N4, that does not belong to the route, the bandwidth consumed by  $f1$  on this node is nearly three times the bandwidth requested by  $f1$ . Indeed, N4 is in the transmission range of N0 and N1 and in the interference area of N2. These three nodes belong to the route of flow  $f1$ . Consequently, N4 is disrupted each time one of these nodes transmit. We conclude that because of the interferences, a flow consumes more bandwidth than it requests. This illustrates the necessity to take into account the interferences in all solutions managing quality of service with bandwidth requirements.

In the following, we assume that interferences caused by a transmitting node are limited to two hops, as usually done.

The maturity of wireless technologies on the one hand, and the evolution of applications on the other hand, justify the introduction of Quality of Service (QoS) in ad-hoc networks. The majority of Quality of Service algorithms used in wired networks relies on the knowledge of precise information concerning the network state. They consider a weak loss rate, a large available bandwidth and a stable network topology. Thus, these algorithms cannot be applied just as they are in a wireless environment. In the following, we present a QoS support designed for ad hoc networks.

## A. QoS Components

In [1] we have presented a general QoS architecture and defined its different components illustrated in Fig.5.

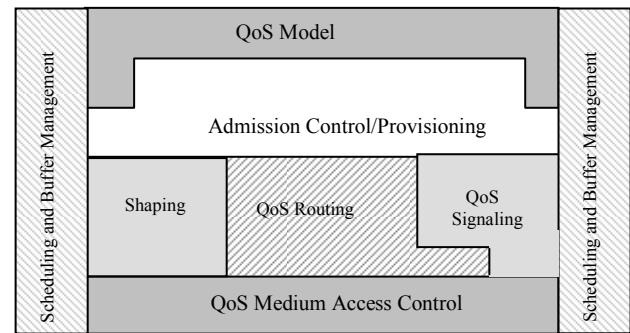


Fig.5. QoS Components

Among these components we are interested in the five following components:

- *QoS model* specifies the architecture in which services can be provided as well as the necessary mechanisms such as classification. The QoS model directly influences the functionality of the other components.
- *Admission control* is the mechanism that allows to accept or reject a new flow according to (i) the available resources on the path taken by this flow and (ii) the QoS requirements of this flow;
- *QoS signaling* is used to propagate QoS control information in the network, to reserve and release resources, as well as to generate the QoS reports that indicate the effectively measured QoS. QoS signaling can be associated with routing with or without QoS.
- *QoS routing* aims to find routes with sufficient resources to meet the application requirements but does not reserve resources.
- *The scheduler* determines the message transmission order according to the priorities given to QoS classes.

### B. WCBQ and Interferences

In this section, we show that WCBQ does not manage interferences. Therefore, the best effort flows can degrade the quality of service of QoS flows already accepted.

Simulations are done with Network Simulator NS2 [8]. Simulation parameters are summarized in the following table:

TABLE I  
SIMULATION PARAMETERS

Simulation	- Duration: 300s - Number of nodes: 8 - Flat area: 1000mx1000m - Traffic type: CBR - Packet size: 500kbytes
OLSR Routing Protocol	- Hello period: 2s - TC period: 5s
MAC	- MAC protocol: IEEE802.11b - Throughput: 2Mb/s - No RTS/CTS messages
Radio	- Radio propagation model: TwoRayGround - Transmission range: 250m - Interference range: 500m

For WCBQ, we have calculated the weight  $\phi(f_j)$  associated with each flow  $f_j$ , present on the node  $N$  and requesting  $B(f_j)$  bandwidth units, as follows:

$$\phi(f_j) = \frac{B(f_j)}{\sum_{i=1 \dots n} B(f_i)}$$

where  $n$  is the number of flows, having the same priority, present on the node  $N$ .

Now, let us consider the following scenario:

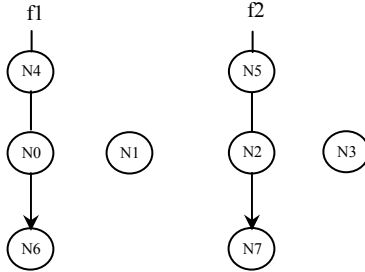


Fig.6. Scenario of 8 nodes and 2 flows

We consider an ad-hoc network constituted by eight nodes (N0, ..., N7). We assume that each flow corresponds to one traffic class and all flows start and stop transmitting at the same time.

First, we consider two QoS flows having bandwidth requirements. The following table specifies source, destination and requested bandwidth for each flow.

TABLE II  
FLOW PARAMETERS

Flows	Requested Bandwidth	Source	Destination
f1	300 kb/s	N4	N6
f2	400 kb/s	N5	N7

We measure the average bandwidth obtained by each flow at its destination node. Fig.7, shows that each flow has received its requested bandwidth (300kb/s for  $f_1$  and 400kb/s for  $f_2$ ).

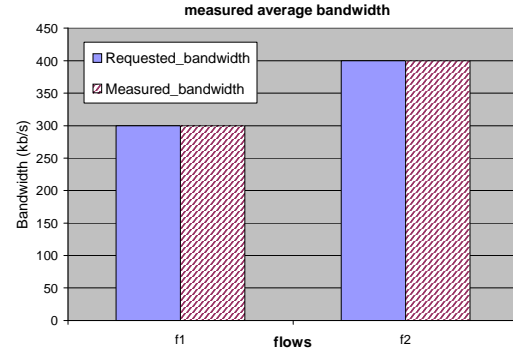


Fig.7. Measured average bandwidth of  $f_1$  and  $f_2$

Now, we introduce a best effort flow  $f_0$  (Fig.8) where its source is N0, its destination is N1 and its requested bandwidth is 200kb/s.

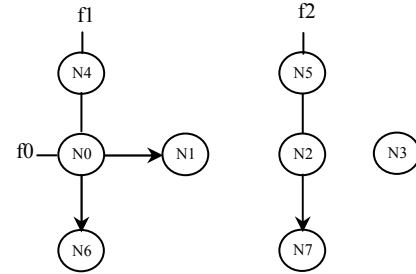


Fig.8. Scenario of 8 nodes and 3 flows

We measure the average bandwidth of each flow at its destination node in two cases:

*Case1:* QoS and BE flows have the same priority

Simulation results (Fig.9) show that (i) the bandwidth received by QoS flow  $f_2$  has fallen from 400kb/s (in absence of  $f_0$ ) to 337kb/s (in presence of  $f_0$ ). (ii) the bandwidth received by QoS flow  $f_1$  has decreased from 300kb/s (in absence of  $f_0$ ) to 291kb/s (in presence of  $f_0$ ). On the other hand, Best effort flow  $f_0$  has received its requested bandwidth (200kb/s).

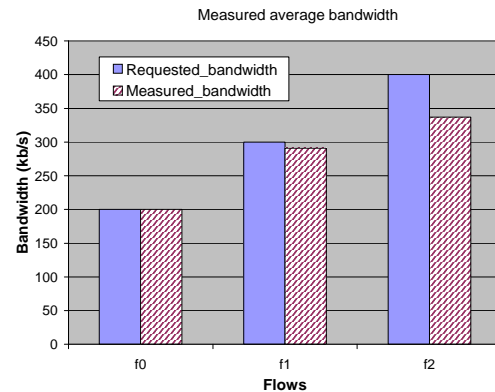


Fig.9. Measured average bandwidth of  $f_0$ ,  $f_1$  and  $f_2$

We notice that the introduction of the BE flow  $f_0$  has generated interferences. These interferences have disrupted QoS flows  $f_1$  and  $f_2$  leading to the degradation of their quality of service.

*Case2:* QoS flows have a higher priority than BE flows

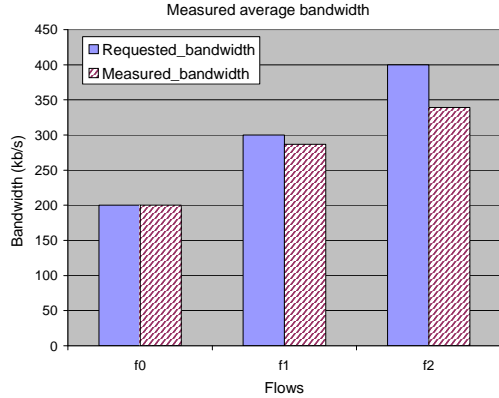


Fig.10. Measured average bandwidth of  $f_0$ ,  $f_1$  and  $f_2$

In this case, though QoS flows  $f_1$  and  $f_2$  have a higher priority than best effort flow  $f_0$ , the quality of service of QoS flow has been degraded by the introduction of BE flow  $f_0$ . Indeed, the bandwidth received by QoS flow  $f_2$  has fallen from 400kb/s in absence of  $f_0$ , to 339kb/s in presence of  $f_0$ . Similarly, the bandwidth received by QoS flow  $f_1$  has decreased from 300kb/s in absence of  $f_0$ , to 287kb/s in presence of  $f_0$ . On the other hand, best effort flow  $f_0$  has received its requested bandwidth (200kb/s).

We can conclude that WCBQ shares the available bandwidth between flows according to their weight without taking into account the interferences generated by flows. Consequently, an admission control is necessary not only for QoS flows but also for best effort flows to take into account the interferences they can generate on QoS flows and to protect QoS flows already accepted.

#### IV. SOLUTION FOR QoS SUPPORT

In this section we present our solution of QoS support taking into account interferences generated by flows present in the network. This solution is made up of the following components: QoS model, Admission control, QoS routing, QoS signaling and scheduling.

First, we present the flow types considered in our solution and show how they are managed. Then we present the admission control rules of a new flow taking into account the interferences generated by this flow. After that, we describe the extension of the routing protocol OLSR in order to provide routes satisfying the QoS requirements. Finally we show how to make coexist different flow types through the WCBQ scheduler.

##### A. QoS Model

We consider two flow types:

- QoS flows having QoS requirements expressed in terms of bandwidth,
- Best Effort (BE) flows having no specific QoS requirements.

To share the medium bandwidth between QoS flows and BE flows, we will use a provisioning. The provisioning consists in reserving a percentage of the nominal bandwidth to each flow type. We consider then:

- $ProvQoS^N$ : provisioning of QoS flows on node  $N$ .
- $ProvBE^N$ : provisioning of BE flows on node  $N$ .

We assume that  $ProvQoS^N$  and  $ProvBE^N$  are global parameters of the network and they are identical on all network nodes. Two solutions can be proposed for the management of QoS and BE flows whether we authorize or not each flow type to exceed its provisioning:

- 1) Each of the two flow types can never exceed its provisioning. In this case, QoS flows consume a bandwidth lower or equal to the QoS provisioning, and BE flows consume a bandwidth lower or equal to the BE provisioning.
- 2) Each of the two flow types can exceed its provisioning. In this case, the bandwidth not used by one flow type can be used by the other, and when necessary, each flow type can recover its share of bandwidth used by the other one.

For an effective use of the node resources, we adopt the second solution *i.e.* each flow type can exceed its provisioning.

Let us consider the following notations:

$BQoS_a^N$ : available QoS bandwidth on node  $N$ .

$BBE_a^N$ : available BE bandwidth on node  $N$ .

$BQoS_u^N$ : QoS bandwidth used on node  $N$ .

$BBE_u^N$ : BE bandwidth used on node  $N$ .

$ProvQoS^N$ : provisioning granted to QoS flows on node  $N$

$ProvBE^N$ : provisioning granted to BE flows on node  $N$ .

We can distinguish three cases of bandwidth consumption of QoS and BE flows (Fig.11).

*Case 1:* no flow type exceeds its provisioning.

*Case 2:* BE flows use all their provisioning and a part of QoS provisioning.

*Case 3:* QoS flows use all their provisioning and a part of BE provisioning.

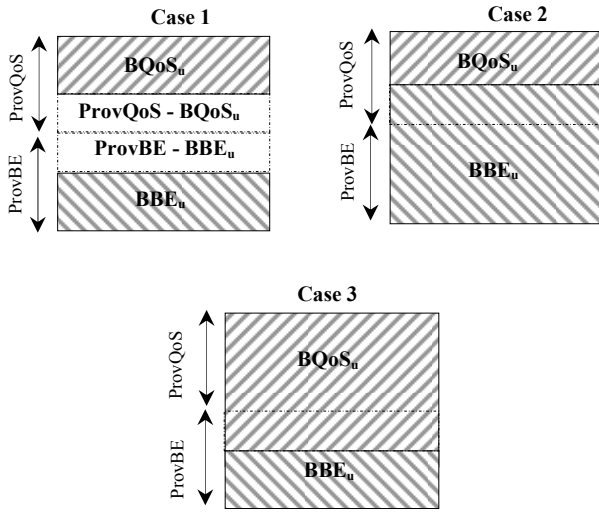


Fig.11. Bandwidth consumption of QoS and BE flows

In our solution, only QoS flows can recover their available bandwidth used by BE flows. BE flows must not recover their available bandwidth used by QoS flows, to avoid the deterioration of the quality of service of QoS flows already admitted. However, if a new QoS flow arrives when the QoS available bandwidth it needs entirely or partially, is used by the BE flows, this flow can recover the bandwidth it needs from BE flows.

### B. Scheduling

For the scheduling, we use WCBQ. As in case 2 of section III, QoS flows have a higher priority than BE flows. The highest priority is granted to the CTRL\_Queue associated with control flows (e.g. OLSR packets).

### C. Admission control

Let us recall that, the admission control is the mechanism deciding whether a new flow is accepted or not, depending on (i) the available resources in the network and (ii) the requirements of this QoS flow. Then, whatever the QoS required by a new flow  $f$ , this flow is accepted by the admission control if and only if:

- the QoS of already accepted QoS flows is not compromised;
- the QoS required by the flow  $f$  can be satisfied.

#### 1) QoS flows with bandwidth requirements

In our solution, the admission control of QoS flows having bandwidth requirements takes into account the interferences *i.e.* a flow will be accepted only if the interferences that it generates are acceptable for already accepted flows and the QoS it will receive is compatible with that required taking into account the interferences generated by other flows. For that, the admission control must check for each route node that the QoS flow is supported by this node and by all its one and two hop neighbors. As the destination does not retransmit the flow

which is intended to it, the admission control, checks only that the flow is supported by its destination. We present the admission control rules in section D.

#### 2) BE flows

BE flows have no specific QoS constraint, but an admission control is necessary to verify that they do not exceed their available bandwidth. The admission control of a new BE flow is carried out locally on each route node. It consists in checking, on each route transmitting node, that the new flow is supported by this node and by all its one and two-hop neighbors.

#### 3) Computation of the needed bandwidth

We note that, because of the interferences, a flow  $f$  requiring a bandwidth  $B(f)$  at the application level, consumes really a bandwidth  $B_{real}(f)$  at the MAC level, higher than  $B(f)$ . This is true on any route node and on any neighbor node of a route node. That is due to the interferences. Before presenting the admission control rules, we show below, how to evaluate the bandwidth really consumed by a flow.

In our solution, the route is supposed to be straight or with a low curvature, so that a route node belongs to the interference zone of itself, at most its two predecessors and at most its two successors. Hence, the value of 5 in formula (1).

$$B_{real}(f) \leq coef.min(5, hop).B(f) \quad (1)$$

Where:

*hop* the number of hops from the source to the destination.  
*coef* a coefficient allowing to take into account the overhead induced by the MAC acknowledgement and the headings of the protocols: physical, MAC, IP and UDP. The *coef* also depends on the packet size. For example, for a QoS flow whose packet size is equal to 500 bytes, and with a medium of 2Mb/s, the value of *coef* is equal to 1.144.

We note that the value  $coef.min(5, hop).B(f)$  corresponds to the maximum bandwidth that a flow can consume on a node *i.e.* the bandwidth really consumed by a flow on any node is never higher than  $coef.5.B(f)$  with our assumptions.

#### 4) Admission control rules

After having calculated the bandwidth really consumed by a flow, we present below the rules of the admission control.

The admission control is performed for the two flow types QoS and BE. It consists in checking:

- For each route node  $N$  (except the destination) and for each node  $M$  at a distance lower than or equal to two hops of  $N$  :  
 - For a QoS flow  $f$  with bandwidth constraints:

$$B_{real}^N(f) \leq BQoS_a^N$$

$$B_{real}^M(f) \leq BQoS_a^M$$



- For a BE flow  $f$

$$B_{real}^N(f) \leq BBE_a^N$$

$$B_{real}^M(f) \leq BBE_a^M$$

• For the destination node  $D$

- For a QoS flow  $f$  with bandwidth constraints:

$$B_{real}^D(f) \leq BQoS_a^D$$

- For a BE flow  $f$

$$B_{real}^D(f) \leq BBE_a^D$$

Where :

$$BQoS_a^N = \max(ProvQoS^N - BQoS_u^N, available^N)$$

$$BBE_a^N = \max(ProvBE^N - BBE_u^N, available^N)$$

$$Available^N = (ProvQoS^N - BQoS_u^N) + (ProvBE^N - BBE_u^N)$$

We will show in the IV.4.B how to calculate the available QoS bandwidth  $BQoS_a$  as well as the available best effort bandwidth  $BBE_a$ .

#### D. QoS Routing

The OLSR routing protocol with QoS support aims at finding:

- for QoS flows, the shortest route satisfying the requested bandwidth.
- for BE flows, the shortest route.

The OLSR extension which we propose consists in: (i) modifying the choice of the multipoint relay and (ii) adding information in control messages Hello and TC, information necessary to the admission control and the QoS routing. We also present the rules of admission control adapted to this extension.

##### 1) Selection of MPRs according to the available bandwidth

In an ad hoc network, the native OLSR protocol provides an optimal route to any destination in the network. This route is optimal in terms of number of hops but does not take into account the requirements of QoS flows. For a QoS flow, we need to find a route which satisfies the required quality of service. However, the route found by OLSR consists of MPR nodes. This is why we perform the MPR selection according to the QoS local available bandwidth denoted  $BQoS_a$ .

In the extension we propose, multipoint relays are selected so as to reach the two hop neighbors through a one-hop neighbor with the maximum QoS available bandwidth ( $BQoS_a$ ) i.e. if a two-hop neighbor can be reached by several one-hop neighbors then the one having the larger  $BQoS_a$  is selected. Because we have taken into account the bandwidth to select the MPR nodes, the MPRs are called MPRBs.

##### 2) Evaluation of the available bandwidth for QoS and BE

The knowledge of the QoS local available bandwidth ( $BQoS_a$ ) is necessary for the MPRB selection as well as for the admission control of QoS flows. On the other hand, the knowledge of best effort local available bandwidth ( $BBE_a$ ) is necessary for the admission control of BE flows. We calculate the  $BQoS_a$  and the  $BBE_a$  on a given node  $N$  as follows:

$$BQoS_a^N = \max(ProvQoS - BQoS_u^N, available^N)$$

$$BBE_a^N = \max(ProvBE - BBE_u^N, available^N)$$

where:  $available^N = (ProvQoS - BQoS_u^N) + (ProvBE - BBE_u^N)$ .

The QoS available bandwidth (or BE available bandwidth) is calculated according to the QoS used bandwidth ( $BQoS_u$ ) and the BE used bandwidth ( $BBE_u$ ). The  $BQoS_u$  (or  $BBE_u$ ) is calculated according to the QoS load and the BE load.

##### • Evaluation of the load for QoS and BE

We define the QoS load ( $QoS\_ch$ ) on a given node  $N$  during a time interval  $T$  as the sum of QoS bytes transmitted by the node  $N$  during  $T$ . We note that QoS bytes include bytes of QoS flow data and bytes of control traffic.

$$QoS\_ch = \frac{\sum QoS\_bytes}{T \cdot MC}$$

Where  $MC$  is the medium capacity.

We also define the BE load ( $BE\_ch$ ) on a given node  $N$  during a time interval  $T$  as the sum of BE bytes transmitted by the node  $N$  during  $T$ .

$$BE\_ch = \frac{\sum BE\_bytes}{T \cdot MC}$$

Each node broadcasts this information ( $QoS\_ch$  and  $BE\_ch$ ) in the Hello messages. Consequently, each node in the ad-hoc network knows the loads  $QoS\_ch$  and  $BE\_ch$  of all its one and two-hop neighbors.

##### • Evaluation of the bandwidth used by QoS and BE

The QoS used bandwidth (or the BE used bandwidth) on a given node  $N$  is equal to the QoS (or BE) load on  $N$  plus the sum of QoS (or BE) loads on the one or two hop neighbor nodes of  $N$ :

$$BQoS_u^N = (QoS\_ch^N + \sum_V QoS\_ch) \cdot coef \cdot MC$$

$$BBE_u^N = (BE\_ch^N + \sum_V BE\_ch) \cdot coef \cdot MC$$

Where:

$V$ : the one and two hop neighbor set of node  $N$

$MC$ : Medium capacity

$Coef$ : a coefficient depending on packet size. It takes into account the overhead generated by MAC acknowledgement and protocol headers: physical, MAC, IP and UDP. The  $coef$  value is identical to that used for the evaluation of the really consumed bandwidth by a flow.



### 3) Modification of Hello and TC messages

We have extended the Hello and TC messages in order to convey the necessary information for QoS routing and admission control. A Hello message, sent by a node, contains the following information:

- its address, its  $QoS\_ch$ , its  $BE\_ch$ , its  $BQoS_a$  and its  $BBE_a$ .
- the address, the  $QoS\_ch$ , the  $BE\_ch$ , the  $BQoS_a$  and the  $BBE_a$  of any one hop neighbor with the link status.

From the Hello messages, each node in the network can know the  $BQoS_a$  of all its one and two hop neighbors. Thus, each node can select its MPRB set.

A TC message contains the following information:

- address of the TC sender,
- $BQoS_a$  of the TC sender,
- $BQoS_{min}$  which correspond to the minimum  $BQoS_a$  of all the one and two hop neighbor of the TC sender,
- Address of the MPRB selectors,
- $BQoS_a$  of the MPRB selectors.

From the received TC messages, each node builds its topology table.

### 4) Route Selection

From its neighbor and topology tables, each node builds its routing table using Dijkstra algorithm. The intermediate nodes of routes toward each destination are MPRB nodes.

- *Route selection for QoS flows*

When a new QoS flow  $f$  is generated on a source node, this source node selects the shortest route offering the demanded QoS by applying Dijkstra algorithm on a copy of the topology and the neighbor tables in which only nodes offering the demanded QoS are present.

The admission control of a new QoS flow is performed on the source node. According to the information it maintains from Hello and TC messages, the source cannot verify correctly the second condition of admission control seen in section IV.2.D because it does not know the  $BQoS_a$  of all neighbors at one and two hops of each node belonging to the route.

In our solution, a QoS flow  $f$  is accepted if and only if for each node  $N$  on the route,  $f$  is supported by (i) the node having  $BQoS_{min}$  which corresponds to the minimum of  $BQoS_a$  of all one and two hop neighbors of node  $N$  and (ii) the node undergoing the maximum of interferences generated by  $f$  i.e. the node having  $B^{max}_{real}(f)$  which corresponds to the maximum of  $B_{real}(f)$  on all one and two hop neighbors of  $N$ .

So, for this solution, the admission control of a new QoS flow  $f$  consists in checking on the source node and for each route node  $N$  (except the destination):

$$B^N_{real}(f) \leq BQoS_a^N$$

$$B^{max}_{real}(f) \leq BQoS_{min}$$

On the destination node  $D$  checking that:

$$B^D_{real}(f) \leq BQoS_a^D$$

If the flow is not accepted on one of the route nodes or on one of the neighbors of one of the route nodes, the flow is rejected. Else, when the route satisfying the requested QoS is found, it will be fixed in order to perform source routing i.e. the list of node route addresses will be included in the header of flow packets. In this way, all packets of this flow will follow the same route to reach the destination. This route is recalculated periodically to verify if there exists either a shorter route satisfying the QoS or a broken link.

- *Route selection for BE flows*

Best effort flows are routed hop by hop and the admission control of these flows is performed locally on each route node and for each packet. Hence, when a new BE flow  $f$  is generated on a source node, this source node checks for each packet, if the destination node exists in its routing table. If the destination does not exist, the packet is rejected. Else the node performs a local control admission for this packet to verify if the flow is supported by itself and by all its one and two hop neighbors. If so, the flow is transmitted toward the next node according to the routing table. We note that, for each packet of a new BE flow  $f$ , the admission control consists in verifying on each route node  $N$  (except the destination):

$$B^N_{real}(f) \leq BBE_a^N$$

$$B^{max}_{real}(f) \leq BBE_{min}$$

$BBE_{min}$  is the minimum available bandwidth for BE flow in the one and two hop neighborhood of  $N$ . It is computed from  $BBE_a$  values received in the Hello messages.

## V. PERFORMANCE EVALUATION

In this section we report performance evaluation of the QoS support described in the previous section. We compare the obtained performances with our solution and those obtained by native OLSR. For this purpose we use the NS2 simulator with parameters given in table I of section III. However, the number of nodes is now 50. We consider two QoS flows ( $f1$  and  $f2$ ) which receive their requested QoS and then, we introduce eight best effort flows ( $f3, \dots, f10$ ). After that, we measure the average bandwidth received by each flow at its destination node.

Nodes distribution in the flat area is given by Fig.12, and flows characteristics are given in table III.

The provisioning on any node for QoS and BE flows is 1400kb/s and 600kb/s respectively. The load is computed on each node every 2 seconds.

TABLE III  
FLOW PARAMETERS

Flows	Type	Requested Bandwidth (kb/s)	Source	Destination
f1	QoS	120	37	3
f2	QoS	140	12	27
f3	BE	50	29	19
f4	BE	50	29	19
f5	BE	50	29	19
f6	BE	50	29	19
f7	BE	50	29	19
f8	BE	50	29	19
f9	BE	50	29	19
f10	BE	50	29	19

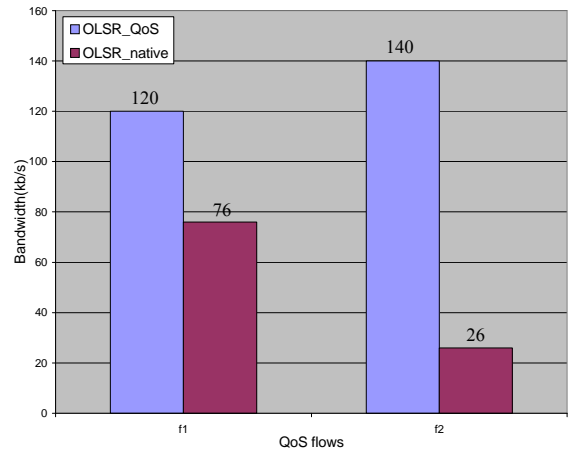


Fig.13. Average measured bandwidth with QoS support and native OLSR

Figures 14 and 15 represent the obtained instantaneous bandwidth received by each flow with respectively QoS support and native OLSR. With native OLSR the instantaneous bandwidth obtained by each flow is very chaotic. Whereas, with QoS support it has weak oscillations around the requested bandwidth.

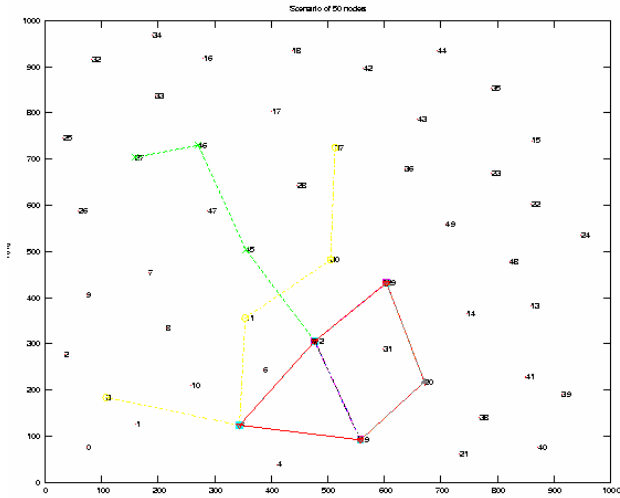


Fig.12. Node distribution in a flat area of 1000m\*1000m

Simulation results show that QoS flows have a QoS close to that requested despite the introduction of BE flows. Their requested and obtained bandwidths are depicted in Fig.13.

Indeed, QoS flow *f1* has obtained 120kb/s, QoS flow *f2* has obtained 140kb/s. On the other hand, BE flows *f3*,...*f10* have shared the available bandwidth on the source node *N29* and each node has obtained 10kb/s.

We have also measured the bandwidth received by each flow using native OLSR (see Fig.13). We can notice that the quality of service received by QoS flows using QoS support is better than using native OLSR: see for instance the throughput granted to QoS flow *f2*, it is equal to 26kb/s with native OLSR and 140kb/s with our QoS support.

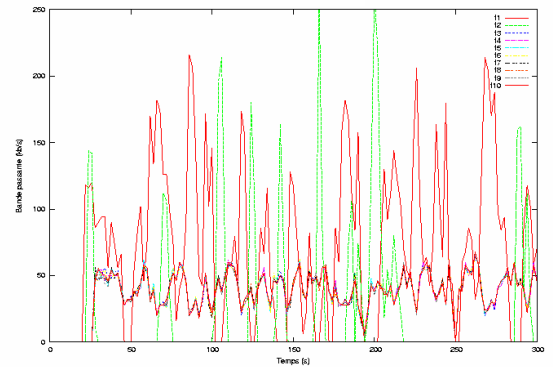


Fig.14. Instantaneous measured bandwidth with native OLSR

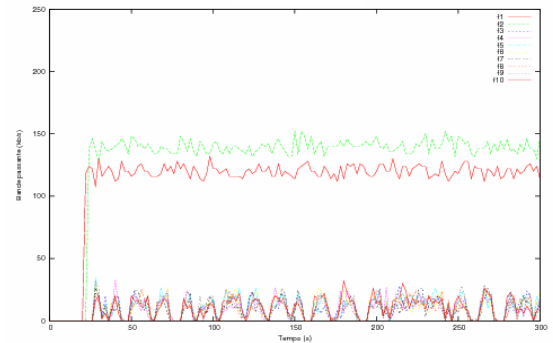


Fig.15. Instantaneous measured bandwidth with QoS support

## VI. CONCLUSION AND PERSPECTIVES

In this paper we have proposed a new QoS support for mobile ad-hoc network taking into account radio interferences. The OLSR routing protocol has been extended for QoS signaling and QoS routing. An admission control has been integrated. We can notice that the QoS support does not require any additional OLSR message. Therefore, the overhead in OLSR message sent by a node each second is kept reasonable. Simulation results show that the accepted QoS flows received the requested throughput. Their QoS is not degraded by the introduction of BE flows.

In a further work we will extend our solution to support flows with delay constraint. We will also show that our solution supports node mobility.

## REFERENCES

- [1] L. Boukhalfa, P. Minet, L. George, S. Midonnet. "Mobile ad-hoc networks and QoS demanding applications", in: 5th IEEE int. conf. on Mobile and Wireless Communications Networks, MWCN'03, Singapore, October, 2003.
- [2] L. Boukhalfa, P. Minet, S. Midonnet, L. George, "Comparative evaluation of CBQ and PriQueue in a MANET", in IEEE int. Workshop on Heterogeneous Multi-hop Wireless and Mobile Networks, IEEE MHWMN'05, Washington, November, 2005.
- [3] Z.Y. Demetrios, "A glance at Quality of Service in Mobile Ad-Hoc Networks" Research report for CS260-seminar in Mobile Ad-Hoc Networks, 2001. Available: <http://www.cs.ucr.edu/~csyiazti/cs260.html>
- [4] S. Floyd and V. Jacobson. "Link-sharing and Resource Management Models for Packet Networks". IEE/ACM transactions on networking, 3(4):365-386, August 1995.
- [5] C. Fragouli, V. Sivaraman, M. Srivastava, "Controlled multimedia wireless link sharing via enhanced class-based queueing with channel state-dependent packet scheduling", in Proc. IEEE INFOCOM'98, pp. 572-577, 1998.
- [6] T. Clausen, P. Jacquet, "Optimized Link State Routing Protocol (OLSR)", Request for Comments RFC 3626, Project Hipercom, INRIA, October 2003.
- [7] J. Moy, "OSPF OSPF Version 2", IETF RFC 2328, April 1998.
- [8] The VINT Project. "The network simulator - ns-2 v.2.1b7a". Available: <http://www.isi.edu/nsnam/ns/>.
- [9] K. Al Agha, G. Pujolle, G. Vivier, "Réseaux de mobiles et réseaux sans fil", Eyrolles 2001.