A Solution for QoS Support in Wireless Ad hoc Networks

Leila Boukhalfa[1, 3], Pascale Minet[2], Serge Midonnet[1, 3]

[1] *Université Marne la Vallée*
77454 Marne-la-Vallée
France
leila.boukhalfa@univ-mlv.fr

[2] *INRIA, Rocquencourt*
78153 Le Chesnay Cedex
France
pascale.minet@inria.fr

[3] *ESIGETEL*
1 rue du Port de Valvins
77210 Avon, France
serge.midonnet@esigetel.fr

Abstract

Mobile ad hoc networks become more popular as devices and wireless communication technologies are became widespread and ubiquitous. With the expanding range of applications of MANETs, supporting quality of services (QoS) in these networks is becoming a real need. This paper provides a solution for QoS support taking into account radio interferences. We note that, because of the ad hoc networks characteristics, we cannot provide a hard quality of service to QoS flows, but only provide a service differentiation between different flow types. This QoS support is based on the OLSR routing protocol and the CBQ scheduling. Simulation results show that flows with QoS requirements receive the requested bandwidth and Best Effort flows share the remaining bandwidth. Moreover, mobility is supported.

Keywords: MANET, QoS, OLSR, CBQ, routing protocol, quality of service.

1. 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 [4] 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 nodes mobility, c) the power constraints because network nodes can rely on battery power for energy. These MANET specificities make it difficult to achieve QoS in these networks.

OLSR [6] is an optimization of the wired link state routing protocol OSPF [7] for MANET. Its innovation lies in the fact that it uses the MultiPoint Relay (MPR) technique. The MPRs of a node are a subset of its one hop neighbors that enables it to reach (in terms of radio range) all its two-hop nodes. The MPRs technique results in the reduction of the control packet size (each node declares only the links with its one hop neighbors which 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.

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 [3] 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 [5] (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, WCBQ leads to good resource utilization.

In this paper, we show how to take into account radio interferences to provide the bandwidth requested by QoS flows. The remainder of this paper is organized as follows. In section 2, we discuss the impact of interferences on the QoS, and describe the QoS components constituting our solution to support QoS in ad hoc networks. The performance evaluation of our solution is given in section 3, followed by a conclusion in section 4.

2. Proposed solution

In this section we present our solution for QoS support taking into account interferences generated by flows present in the network.

2.1. QoS and interferences

Because of the shared medium access in ad hoc networks, a packet generated by a mobile node is physically received by all nodes in the transmission range of the sender. Consequently, interferences are generated when neighboring nodes are transmitting at the same time. The presence of interferences makes quality of service support much more complex in wireless networks than in wired networks. For example, interferences make bandwidth reservation in a wireless environment an NP-complete problem [9], whereas it is polynomial in wired networks. These interferences can reduce significantly the capacity of the network.

Let us consider a scenario of 6 nodes and one flow f_1. The flow f_1 requests a bandwidth of 100kb/s. f_1 is generated by node N_0 toward node N_3 (Figure 1). To illustrate the interference phenomenon, we measure the consumed bandwidth at the MAC level on each node of the network.

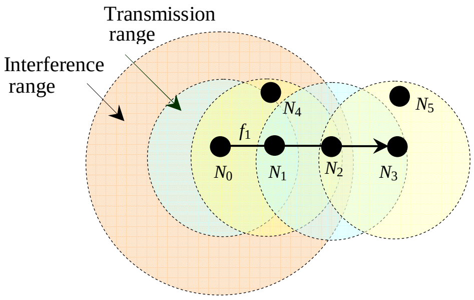

Figure 1. Interference phenomenon

We note that (see Figure 2) flow f_1 has consumed 281kb/s on N_2. It represents nearly three times the bandwidth requested by f_1. Indeed, node N_2 is disrupted by any packet of flow f_1, once when N_0 transmits, because N_2 is in the interference area of N_0, a second time when N_1 transmits because $N2$ is in the transmission range of N_1, and a third time when the node itself transmits. As for node N_5, it does not belong

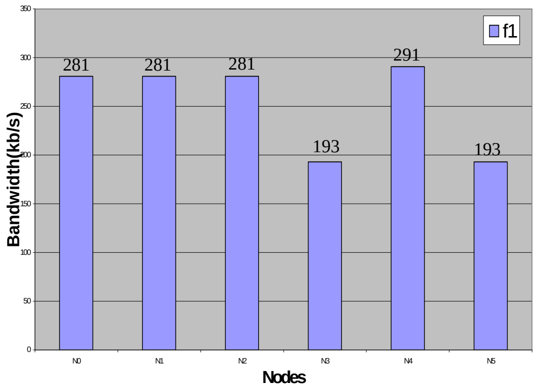

Figure 2. Measured consumed Bandwidth

to the route, the bandwidth consumed by f_1 on this node is nearly twice the bandwidth requested by f_1. Indeed, N_5 is in the interference area of N_1 and N_2. These two nodes belong to the route of flow f_1. Consequently, N_5 is disrupted each time one of these nodes transmits. 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.

Providing quality of service in ad hoc networks therefore should be interference aware [10]. For this goal, we consider an admission control which takes into account interferences induced by flows present on the network. Also, the routing protocol considered in our solution takes into account interferences to provide routes with the requested quality of service. QoS routing needs QoS signaling to collect information related to QoS. Besides these components, other QoS components can be used to provide the quality of service requested by QoS flows. Hence, the QoS architecture we propose in the next section.

2.2. QoS components

In [1] we have presented a general QoS architecture and defined its different components illustrated in Figure 3. 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 results in the acceptance or rejection of a new flow according to *(i)* the available resources on the path taken by this flow and *(ii)* the QoS requirements of this flow.

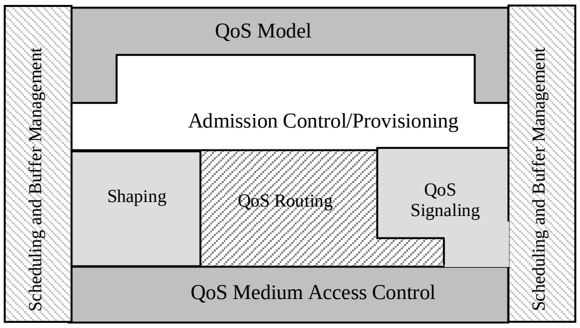

Figure 3. QoS Components

- *Admission control* is the mechanism that results in the acceptance or rejection of 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 routing* aims to find routes with sufficient resources to meet the application requirements but does not reserve resources. In our solution, QoS routing is based on an extension of OLSR.
- *QoS signaling* is used to propagate QoS control information in the network, as well as to generate the QoS reports that indicate the effectively measured QoS.
- *The scheduler* determines the message transmission order according to the priorities given to QoS classes. In our solution, it is based on CBQ.

2.3. Adopted assumptions

2.3.1. QoS MAC. In the ideal case, a medium access protocol managing QoS makes it possible to guarantee, to the non-pre-emptive effect close, that the transmitted packet is the packet having the highest priority among all packets waiting for transmission. Let us notice that the IEEE802.11e protocol does not satisfy this property. It guarantees only that the average delay of flows with higher priority is weaker than that of flows with lower priority.

The MAC layer must be also able to provide information allowing to calculate the available bandwidth on a node. The QoS management on the MAC level allows to obtain a better services differentiation as it shown in [11] for the IEEE 802.11 protocol where flows with higher priority obtain weaker average delays.

The solution we propose does not require a QoS MAC to behave properly. In the performance evaluation reported in section 3, we use the IEEE 802.11b MAC protocol which is more currently used for the MANET networks but not yet offering QoS management. However, the performances of our solution will be improved by a QoS MAC.

2.3.2. Interference model. The proposed solution takes in consideration the interferences to *h* hops:
- at the sender node: the receptions of the nodes located at less than *h* hops of the sender are disturbed when this sender transmits.
- at the receiver node: the simultaneous transmissions of nodes located at less than *h* hops of the receiver prevent the good reception.

Consequently, two senders located at less than 2*h* hops can interfere with each other. The interference range is said to be 2*h*. In this paper we take an interference range 2. This assumption is generally adopted in ad hoc networks literature.

2.3.3. Computation of the needed bandwidth. We note that, because of the interferences, a flow *f* requiring a bandwidth $B(f)$ at the application level, really consumes 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. We show below, how to evaluate the bandwidth really consumed by a flow.

In our solution, the route is supposed to be such that a route node belongs to the interference zone of, at most, its two predecessors and, at most, its two successors; hence the value of 5 in formula (1).

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

Where:
hop is the number of hops from the source to the destination.
coef is 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 which 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.

The formula can appear too simple but any more sophisticated method wanting to take into account all the exact interferences requires a transmission overhead without allowing an exact evaluation as shown in the following example:

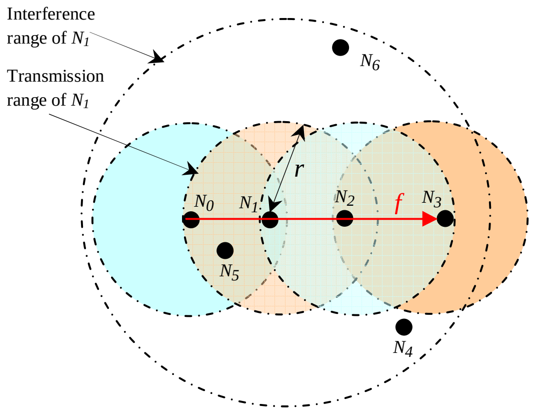

Figure 4. Scenario of 7 nodes

Node N_6 is located in the interference range of node N_1 but not in its transmission range *i.e.*, $r < d(N_1, N_6) < 2r$ where $d(N_1, N_6)$ is the distance between the two nodes N_1 and N_6 and r is the transmission range. In no case, the node N_1 can detect the presence of node N_6, because there is no intermediate node belonging to the transmission range of N_1 and of N_6 making it possible each of the two nodes to detect the presence of the other. Consequently, the evaluation of the bandwidth really consumed by flow f does not consider the disturbances induced by node N_1 on node N_6.

2.4. QoS model

2.4.1. Considered flow types. We consider three flow types:

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

In our solution, we adopt the following decreasing priority order of flows:
Control flows > QoS flows with delay constraints > QoS flows with bandwidth constraints > Best Effort flows.

2.4.2. Bandwidth provisioning. To share the medium bandwidth between QoS flows and BE flows, we will use 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. For an effective use of the network resources, we allow each flow type to 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. Moreover, QoS flows can requisition the bandwidth used by BE flows. The reverse is not true.

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 BE flows, this flow can recover the bandwidth it needs from BE flows.

2.5. Admission control

Let us recall that, the admission control decides to accept a new flow f if and only if:

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

We present below the rules of the admission control. The admission control is performed for the two flow types QoS and BE:

- 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.
- BE flows do not require any constraint, but an admission control is necessary to verify that they do not exceed their available bandwidth.

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.

More particularly, the admission control 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:

❶ $B^{N}_{real}(f) \leq BQoS_a^{N}$

❷ $B^{M}_{real}(f) \leq BQoS_a^{M}$

- For a BE flow f

❶ $B^{N}_{real}(f) \leq BBE_a^{N}$

❷ $B^{M}_{real}(f) \leq BBE_a^{M}$

- For the destination node D

- For a QoS flow f:

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

- For a BE flow f

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

Where:
$BQoS_a^{N} = \max(ProvQoS^{N} - BQoS_u^{N}, available^{N})$
$BBE_a^{N} = \max(ProvBE^{N} - BBE_u^{N}, available^{N})$
$Available^{N} = (ProvQoS^{N} - BQoS_u^{N}) + (ProvBE^{N} - BBE_u^{N})$

2.6. QoS routing

Routing protocol OLSR with QoS 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.

2.6.1. 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 intermediate nodes of a requested route found by OLSR are MPR nodes. This is why we perform the MPR selection according to the QoS local available bandwidth denoted $BQoS_a$.

In the extension that 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.

2.6.2. 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:

$$BQoS_u^{N} = \left(QoS\_ch^{N} + \sum_{V} QoS\_ch \right) \cdot coef.MC$$
$$BBE_u^{N} = \left(BE\_ch^{N} + \sum_{V} BE\_ch \right) \cdot coef.MC$$

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 bandwidth really consumed by a flow.

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

A. 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 the 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 2.5 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 N and (*ii*) by any node up to two hops from N, if N is not the destination.

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. Otherwise, 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.

B. 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. Otherwise the node performs a local control admission for this packet to verify if the flow is supported by this node 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 of verifying on each route node *N* that flow *f* is supported by (*i*) the node *N* and (*ii*) by any node up to two hops from *N*, if *N* is not the destination. This computation is done using BBE_{min}, 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.

2.7. QoS signaling

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.

2.8. WCBQ scheduling

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 [5] (Class Based Queueing), we have extended it to the wireless environment and we have called it WCBQ (Wireless CBQ). WCBQ inherits the three modules of CBQ which are:

- *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.

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 weight.

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.

Let us recall that, in our QoS model, we have considered three user flow types which are QoS flows having QoS requirements expressed in terms of bandwidth, QoS flows having QoS requirements expressed in terms of delay and Best Effort (BE) flows having no specific QoS requirements. To schedule these three user flow types, we have combined the use of two schedulers which are WCBQ and Priority Queueing (PQ) in the following way (see Figure 6):

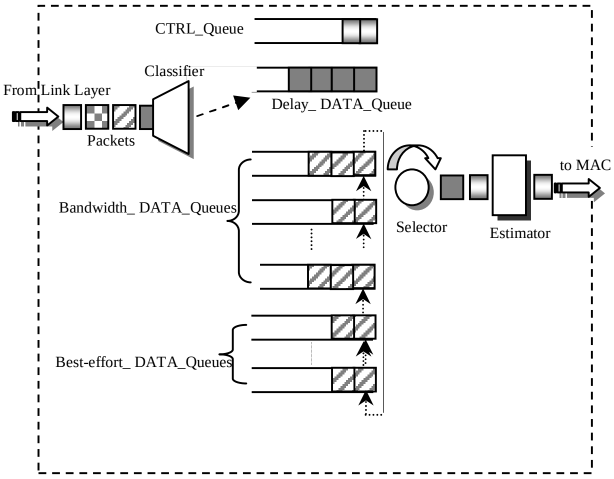

Figure 6. Coexistence of three user flow types

- CTRL_Queue is dedicated to control packets (e.g. routing packets). This queue has the highest priority, thus, it is served first *i.e.*, when a control packet arrives at the CTRL_Queue, either it is transmitted immediately, if there is no packet being transmitted or it is transmitted after the packet in the course of transmission.
- We have also attributed a Delay_DATA_Queue for traffic having delay requirements. This Queue has a lower priority than CTRL_Queue *i.e.*, a packet from Delay_DATA_Queue is transmitted if and only if there is no packet to transmit in CTRL_Queue.
- Bandwidth_DATA_Queues are reserved for traffic having bandwidth constraints.
- Best-Effort_DATA_Queues are reserved for BE flows.

Bandwidth_DATA_Queues and Best-Effort_DATA_Queues are managed according to WRR (Weighted Round Robin). Also, these queues, are served only if there is no packet in CTRL_Queue and no packet in Delay_DATA_Queue for transmission. Thus, with the combination of the two schedulers PQ and WCBQ, we can enable the three user flow types to coexist.

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

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

Where n is the number of flows present on the node N having the same priority as f_j.

3. Performance evaluation

We now report performance evaluation of the QoS support described in the previous section.

3.1. Simulation parameters

The solution performance evaluation is carried out under the NS2 simulator [8]. The network simulator NS2 is an object-oriented, discrete event-driven network simulator. First, we consider an ad hoc network made up of 50 static nodes. The simulation parameters are summarized in the following table:

Table 1. Simulation parameters

Simulation	- simulation duration: 300s - Number of nodes: 50 - Flat area: 1000mx1000m - Traffic type: CBR - Packet size: 500 bytes
Routing protocol (OLSR)	- Source routing for QoS flows - Hop by hop routing for BE flows - Periodic routing table calculation for QoS flows (period 2s) - Hello period: 2s - TC period: 5s - Use of MPRB
MAC	- MAC protocol: IEEE802.11b - Throughput: 2Mb/s - No RTS/CTS messages
Radio	- Radio propagation model : TwoRayGround - Transmission range: 250m - Interference range: 500m

3.2. Fair sharing of bandwidth for BE flows and routes stability for QoS flows

In this section we show that, on a node, BE flows share the available bandwidth proportionally to their weight. In order to do this, we consider six QoS flows (f_1….,f_6) which obtain their requested bandwidth. Afterwards, we gradually introduce ten identical BE flows (f_7....., f_{16}) *i.e.*, same rate, same source and same destination. Each time we measure the bandwidth received by each flow present in the network. We provide also, the number of routes taken by each flow as well as the number of route changes during the simulation. Simulation results are given in Table 3. The source, the destination and the requested bandwidth of each flow are given in Table2.

Table 2. Flows parameters

Flow	Type	Requested bandwidth (kb/s)	Source	Destination
f_1	QoS	50	43	10
f_2	QoS	40	27	48
f_3	QoS	60	18	7
f_4	QoS	30	1	32
f_5	QoS	50	19	28
f_6	QoS	40	41	15
f_7	BE	20	38	12
f_8	BE	20	38	12
f_9	BE	20	38	12
f_{10}	BE	20	38	12
f_{11}	BE	20	38	12
f_{12}	BE	20	38	12
f_{13}	BE	20	38	12
f_{14}	BE	20	38	12
f_{15}	BE	20	38	12
f_{16}	BE	20	38	12

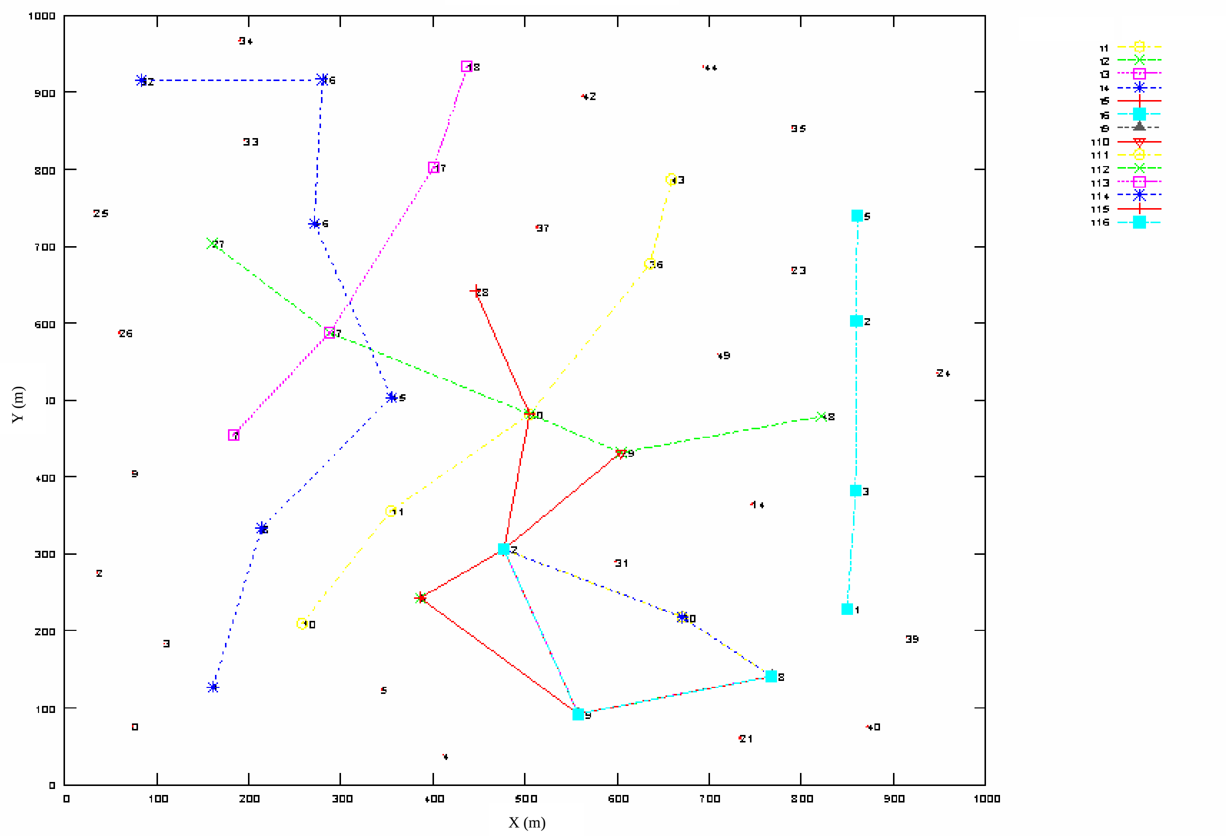

Figure 7. An ad-hoc network of 50 nodes

Table 3. Simulation results

Flows	Type	Requested bandwidth (kb/s)	Measured bandwidth (kb/s)	Route numbers	Number of route changes
f_1	QoS	50	49	1	0
f_2	QoS	40	38	1	0
f_3	QoS	60	60	1	0
f_4	QoS	30	29	1	0
f_5	QoS	50	48	1	0
f_6	QoS	40	40	1	0
f_7	BE	20	4	3	12
f_8	BE	20	4	3	11
f_9	BE	20	4	3	11
f_{10}	BE	20	4	3	11
f_{11}	BE	20	4	3	11
f_{12}	BE	20	4	3	10
f_{13}	BE	20	4	2	8
f_{14}	BE	20	4	3	12
f_{15}	BE	20	4	3	10
f_{16}	BE	20	4	3	10

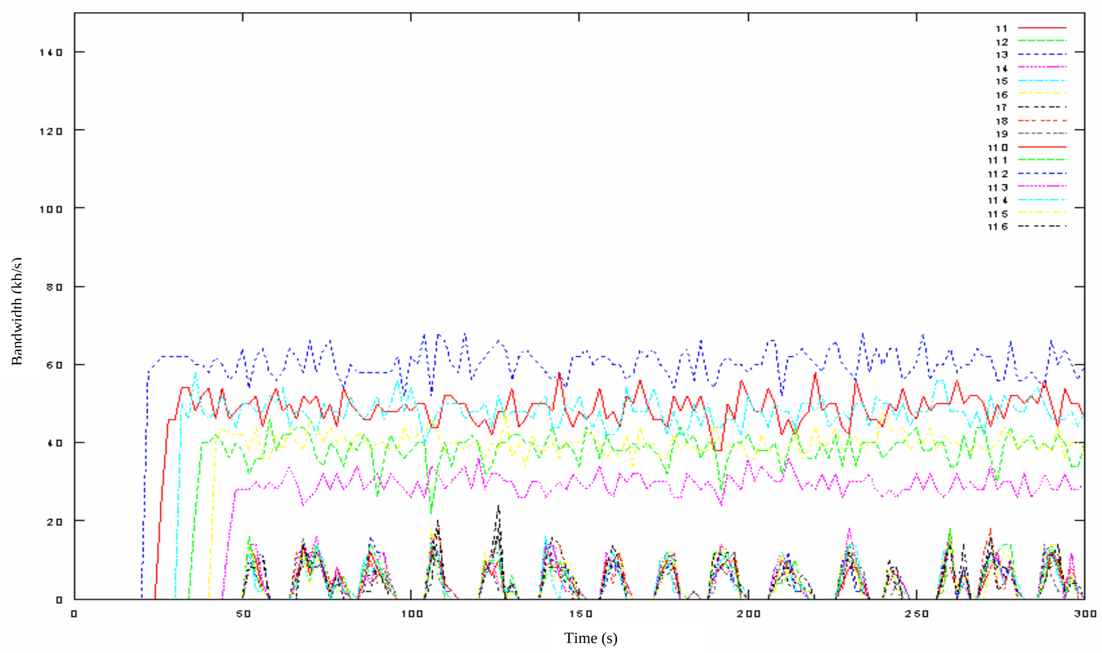

Figure 8. Measured instantaneous bandwidth for 6 QoS flows and 10 BE flows

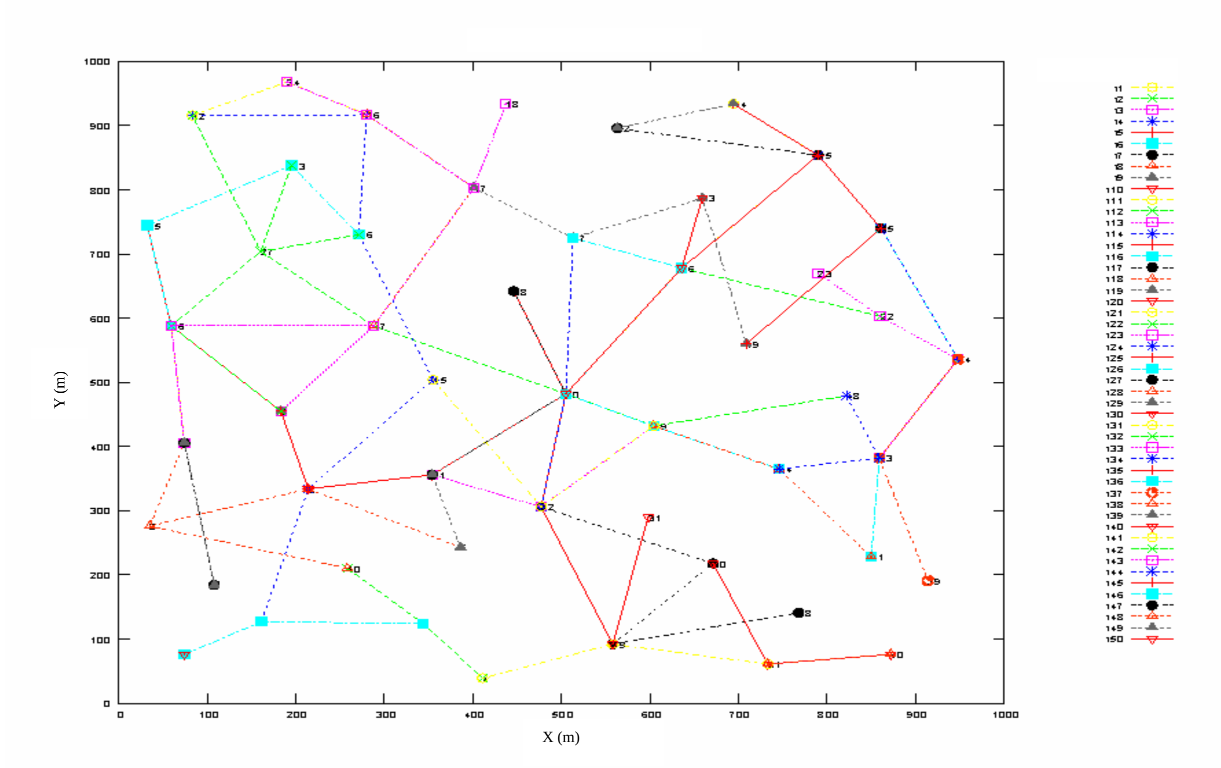

Figure 9. Routes representation of 50 flows

Figure 10. Measured instantaneous bandwidth for 6 QoS and 44 BE flows

According to the simulation results, we can conclude that BE flows share the available bandwidth proportionally to their weight. In the considered scenarios, all BE flows request the same bandwidth and obtain the same weights for WCBQ. They also obtain the same measured bandwidth (Figure 8).

Concerning QoS flows, once admitted, they receive their requested bandwidth whatever the number of BE flows introduced in the network. We now consider another scenario with a higher number of flows where each node in the network generates at least one flow. We consider 50 flows: 6 QoS flows with the parameters given in the above Table 2 and 44 BE flows. Each BE flow requests a bandwidth of 10kb/s. In Figures 9 and 10, we present respectively routes taken by each flow, and the instantaneous bandwidth received by each flow in the network.

The results of the second example confirms that, in spite of the significant number of BE flows present in the network, QoS flows, once admitted, obtain their requested bandwidth. We also notice that the route of QoS flows is much more stable than the route of BE flows. Thus, QoS flows (f_1…,f_6) always use the same route whatever the number of BE flows present. The number of route changes of BE flows is very large, it can reach 12 during a simulation (300s).

3.3. Requisition of bandwidth by QoS flows

In our solution, the routing table of QoS flows is periodically computed in order to provide the shortest route satisfying the requested QoS. To study this characteristic, we consider the network of 50 nodes with at the beginning an overloaded zone with 44 BE flows. The bandwidth requested by each BE flow is equal to 10kb/s. We introduce QoS flows (f_1…, f_6) at times t_1 = 100s, t_2 = 106s, t_3 = 112s, t_4 = 118s, t_5= 124s and t_6 = 130s respectively. Flows f_1 and f_5 request a bandwidth of 60kb/s, flows f_2 and f_4 request a bandwidth of 30kb/s and flows f_3 and f_6 request a bandwidth of 40 kb/s. We stop the transmission of BE flows at time t = 200s, and then, we study the behavior of QoS flows in the absence of BE flows.

Figure 11 represents routes taken by each flow, and Figure 12 represents the instantaneous bandwidth received by each flow.

In conclusion, this configuration shows that the six QoS flows did not circumvent the overloaded zone by BE flows. Indeed, when QoS flows arrive, they requisition the bandwidth used by BE flows. Moreover, each QoS flow takes only one route. This route does not change even in the absence of BE flows, because it is the shortest route satisfying the requested bandwidth. Consequently each QoS flow receives its requested bandwidth.

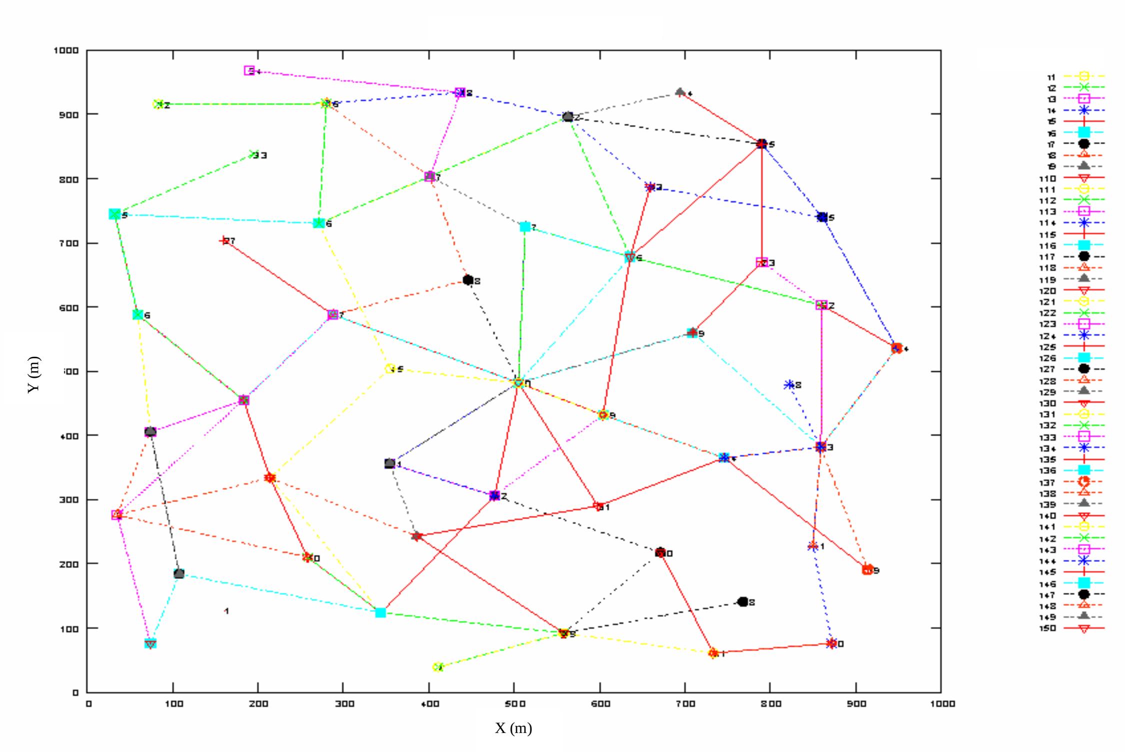

Figure 11. Routes representation

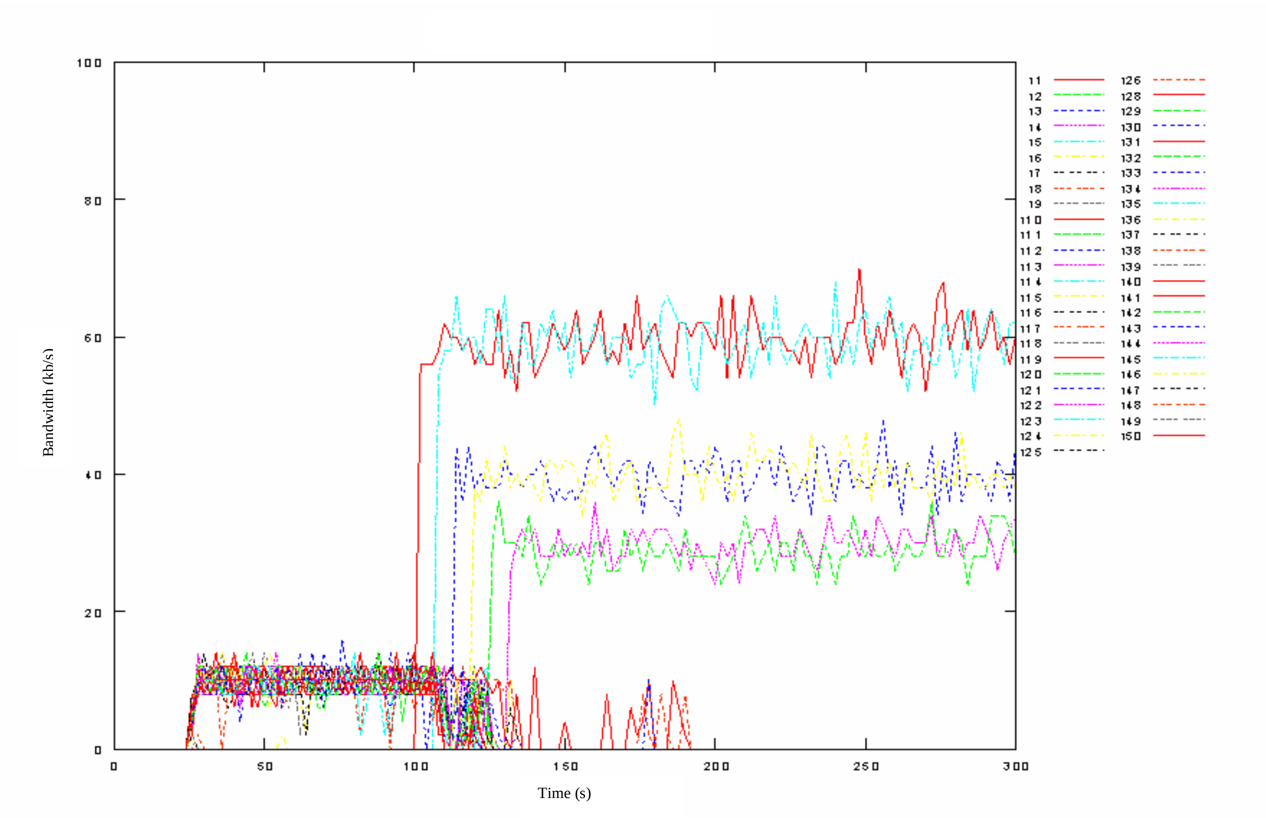

Figure 12. Measured instantaneous bandwidth

3.4. Benefits brought by QoS support

In this section, we evaluate the benefits brought by the QoS support in terms of bandwidth obtained by QoS flows. We consider again the scenario described in the section 3.2, including 6 QoS flows and 10 BE flows. We compare the performances obtained by our solution and those obtained by native OLSR. Figure 13 shows that with the QoS support, each QoS flow obtains the required band-width while with native OLSR, QoS flows f_1, f_2, f_3, f_4, f_5 and f_6 obtain only 28kb/s, 26kb/s, 40kb/s, 24kb/s, 33kb/s and 32kb/s respectively.

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

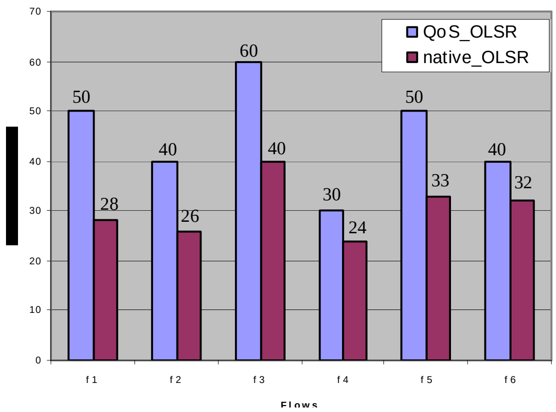

Figure 13. Average measured bandwidth of QoS flows with QoS OLSR and native OLSR

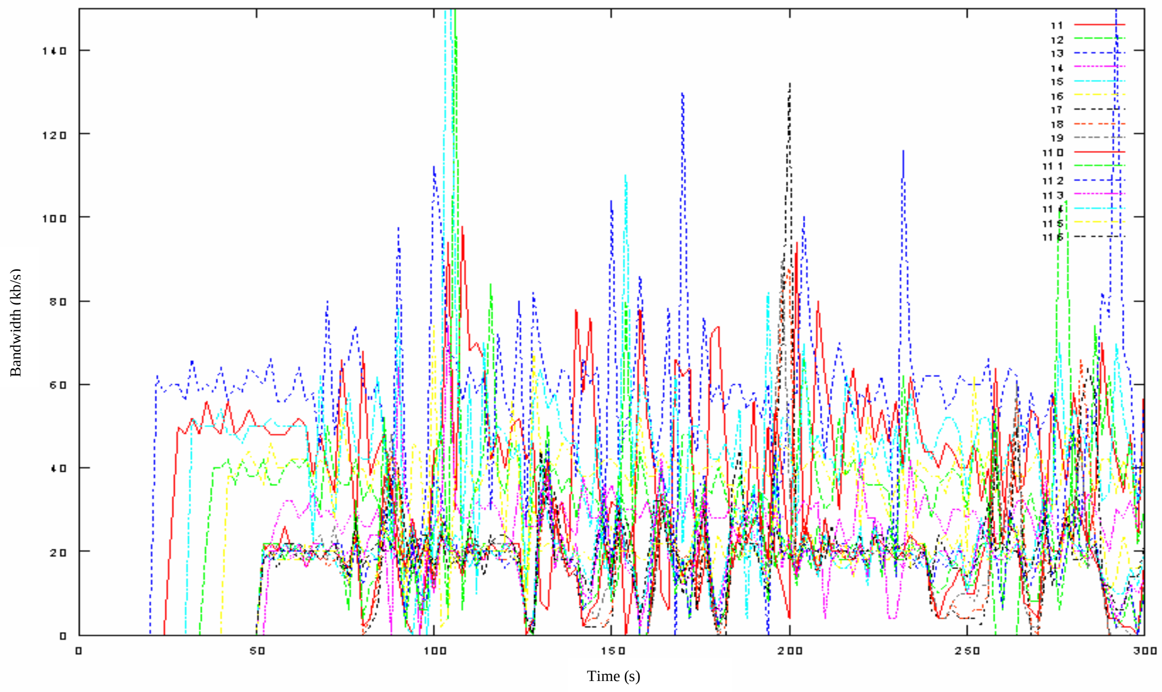

Figure 14. Measured instantaneous bandwidth with native OLSR

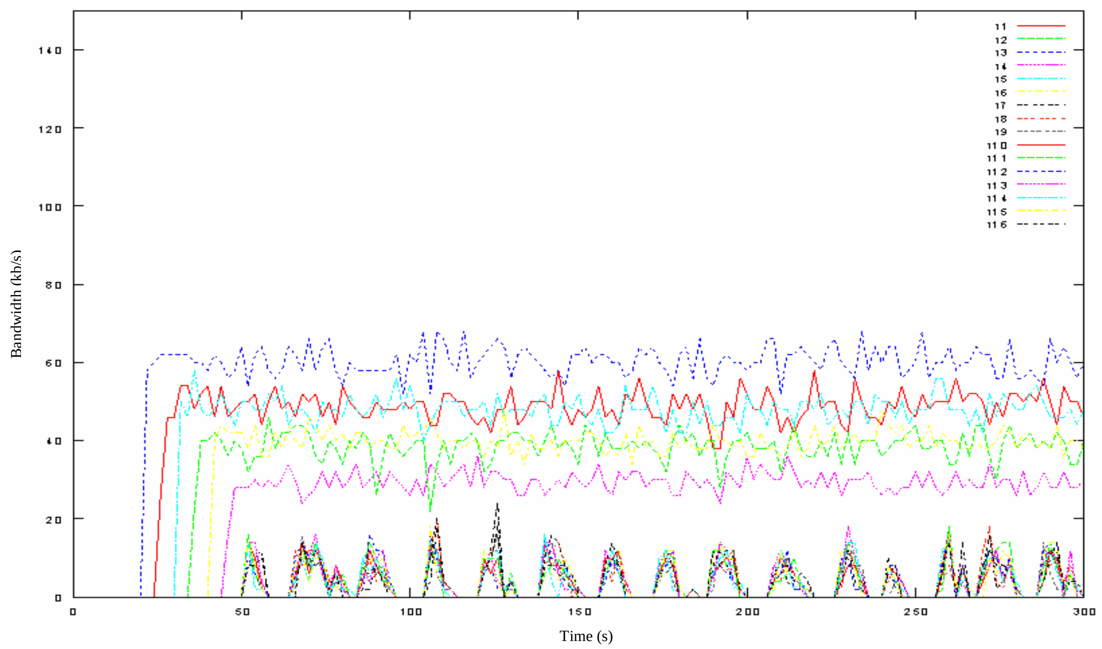

Figure 15. Measured instantaneous bandwidth with QoS support

3.5. QoS flows with delay constraints

In our solution, flows having delay constraints are prioritized compared to flows with bandwidth constraints and BE flows. In this section, we study the interest of the delay flows class and show the interest of having several priorities in this class.

In an ad hoc network of 50 nodes, we consider three flows: a first one with delay constraint (DL), a second one with bandwidth constraint (BW) and a third one with no constraint (BE). The three flows have the same rate (100kb/s), the same source (N_6) and the same destination. According to the number of hops and for each flow, we measure the received bandwidth and the end-to-end delay. These measurements are presented in figures 16 and 17:

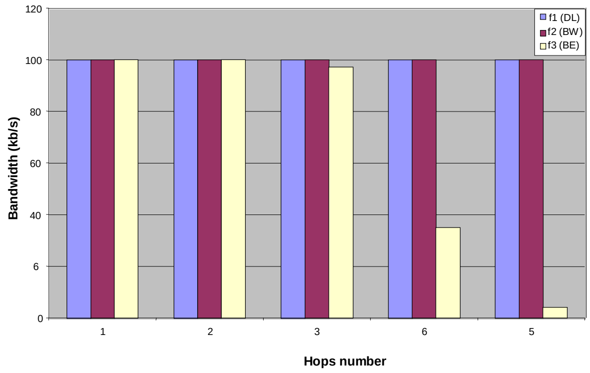

Figure 16. Measured average bandwidth according to hops number

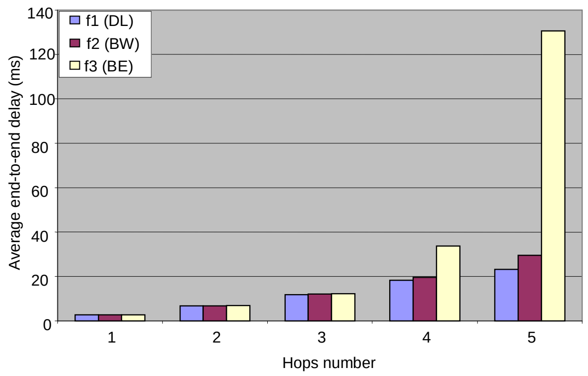

Figure 17. Measured average end-to-end delay according to hops number

We can conclude that once admitted, QoS flows with delay constraint and flows with bandwidth constraint obtain their requested bandwidth. For a number of hops higher than three, the end-to-end delay of flows having delay constraint is smaller than that obtained by flows having bandwidth constraint and that obtained by BE flows. This method thus makes it possible to privilege flows with delay constraint. Consequently, they obtain shorter delays.

Now, let us analyse how to manage Delay_DATA_Queue in the presence of several flows having different delay constraints. For that, we carry out a comparative study between the two schedulers: Priority Queueing (PQ) and Earliest Deadline First (EDF).

- With PQ, the flow having the smallest end-to-end delay receives the highest priority. Packets are inserted in the Delay_DATA_Queue according to their priority *i.e.*, the packet with the highest priority is inserted ahead of the Queue. If there at least one packet with the same priority, the new packet is inserted after the last one. Packets are then transmitted in a FIFO manner.
- With EDF, a local deadline is associated to each packet. This local deadline is calculated according to (*i*) the end-to-end deadline of the flow to which it belongs and (*ii*) the number of hops towards the destination. On a node, packets are inserted in the Delay_DATA_Queue according to their local deadline *i.e.*, the packet with the smallest local deadline is inserted ahead of the Queue. Packets are then transmitted in a FIFO manner. In the following we demonstrate how to calculate deadlines of packets. For that let us define the following notation:

ete_rel_dead: end-to-end relative deadline
ete_abs_dead: end-to-end absolute deadline
loc_abs_ded: local absolute deadline
t: packet generation time on the source node
t_a: packet arrival time on a route node
r: number of hops from the current node towards the destination.

- To each packet generated at time *t* on the source node are associated:

$$ete\_abs\_dead = t + ete\_rel\_dead$$
$$loc\_abs\_ded = t + ete\_rel\_dead / r.$$

- On each route node (except the destination), the local absolute deadline of each packet is recalculated:

$$loc\_abs\_ded = (ete\_abs\_dead - t_a)/r + t_a$$

To compare the two schedulers, we consider five QoS flows having delay constraints. They have also the same source, the same destination and follow the same route made up of four hops. The delay constraints of flows are expressed in term of end-to-end relative deadline. The bandwidth of the 5 flows f_1, f_2, f_3, f_4 and f_5 is 70kb/s, 60kb/s, 60kb/s, 50kb/s and 50kb/s respectively. And their end-to-end relative deadlines are 0.8s, 0.9s, 1s, 1.1s and 1.2s respectively.

In our simulations, we measure, with the two schedulers PQ and EDF and for each flow:(*i*) the average end-to-end deadline (see Figure 18), (*ii*) the maximum end-to-end deadline (see Figure 19), and (*iii*) the rate of packets respecting their deadline (see Figure 20).

According to the simulation results, we can notice that:

- PQ tends to transmit in priority flows with the highest priority. That is why the measured average end-to-end delay and measured maximum end-to-end delay (see Figures 18 and 19) increase when the priority associated to the flow decreases. This strongly impacts the rate of packets respecting their deadline for the flows having the weakest priority (flow f5 in our case).
- EDF, on the other hand, tends to transmit all flows. Indeed, EDF is a dynamic scheduler based on the absolute deadlines. Priority of flows changes according to these absolute deadlines *i.e.*, at a given time, the flow having the smallest absolute deadline is scheduled for transmission. Consequently, EDF, which is known for its scheduling optimality [12] in the single processor context, provides a better rate of packets respecting their deadlines.

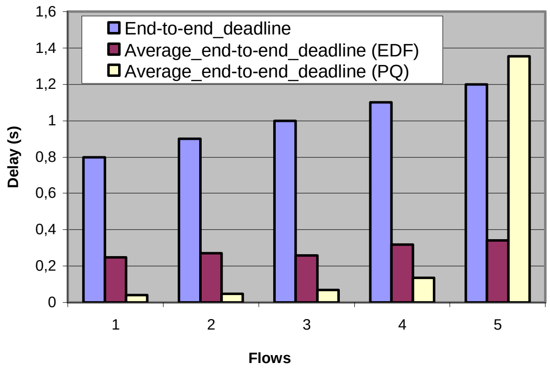

Figure 18. Measured average end-to-end deadline with EDF and PQ

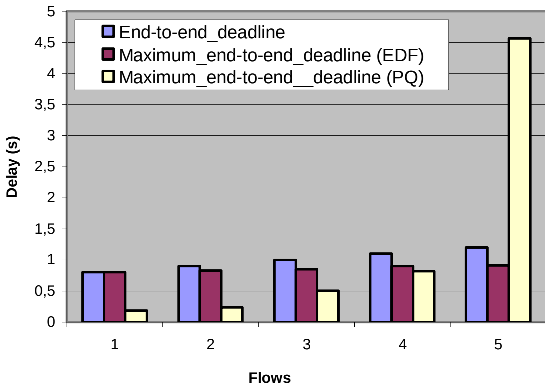

Figure 19. Measured maximum end-to-end deadline with EDF and PQ

According to this simulation results, we recommend to use a dedicated queue for flows having end-to-end delay constraints. We have evaluated the performance of EDF scheduling when the local deadline is computed has the difference between the end-to-end deadline and the time already spent in the network divided by the remaining number of hops towards the destination. Simulation results show that this EDF allows a higher rate of packets meeting their deadline. Therefore, we recommend its use for the Delay_Data_Queue.

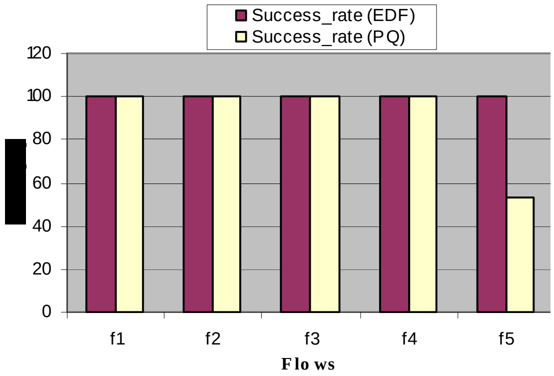

Figure 20. Rate of packets respecting their deadline

3.6. Overhead

Let us evaluate the overhead induced by the QoS support on each node. Thus, for each node, we calculate the number of OLSR messages sent per second. This number takes into account the OLSR messages generated by a node as well as the OLSR messages forwarded by this node. Figure 21 illustrates the overhead calculated for the scenario described in section 3.2 including 10 BE flows and 6 QoS flows.

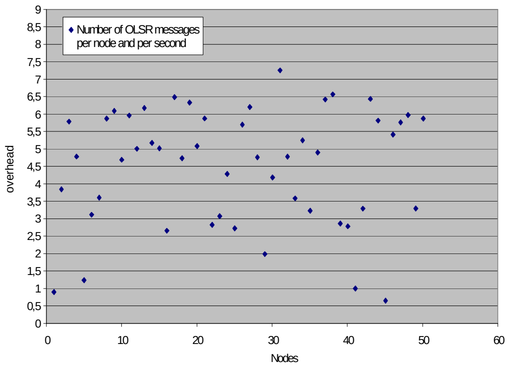

Figure 21. Overhead of QoS support

The average value of the overhead is equal to 4.504. It is larger than the average of the overhead obtained with native OLSR which is equal to 3.445 (see Figure 21). That is due to the fact that with the support of QoS, more nodes are selected as MPRBs and consequently generate TC messages in addition to Hello messages.

3.7. Mobility support

Now, we study the impact of the mobility on the QoS support performances. For the same network, we compare the performances obtained in the presence and the absence of QoS support. This evaluation is carried out on a network of 100 nodes (see Figure 22). The mobility model considered in the simulations is Random Waypoint Model (RWM) where the maximum speed for each node is limited to 5m/s. We define in Table4 the remaining simulation parameters.

In the considered network, 10 flows are present including 2 QoS flows with bandwidth constraint, and 8 BE flows. In table 5, we indicate the requested bandwidth, the source and the destination of each flow.

Table 4. Simulation parameters for an ad hoc network of 100 mobile

Simulation	- simulation duration: 300s - Number of nodes: 100 - Flat area: 1000mx1000m - Traffic type: CBR - Packet size: 500kb
Routing protocol (OLSR)	- Source routing for QoS flows - Hop by hop routing for BE flows - Periodic routing table calculation for QoS flows (period 1s) - Hello period: 1s - TC period: 5s - Use of MPRB
MAC	- MAC protocol: IEEE802.11b - Throughput: 2Mb/s - No RTS/CTS messages
Radio	- Radio propagation model : TwoRayGround - Transmission range: 250m - Interference range: 500m

For each flow present in the network, we measure the received bandwidth, we also provide the number of routes taken as well as the number of route changes during the simulation (see Table 6).

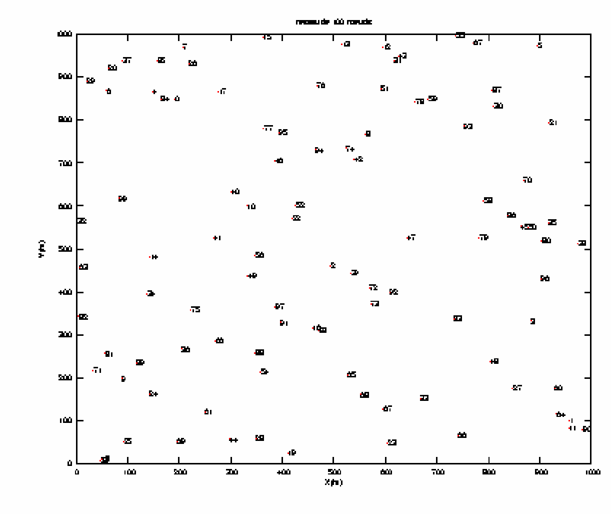

Figure 22. Ad hoc network of 100 mobile nodes

Table 5. Flows parameters

Flow	Type	Requested bandwidth (kb/s)	Source	Destination
f1	QoS	120	67	38
f2	QoS	140	71	8
f3	BE	50	49	58
f4	BE	100	67	90
f5	BE	80	22	0
f6	BE	80	71	19
f7	BE	80	94	61
f8	BE	50	5	66
f9	BE	80	60	72
f10	BE	50	76	30

Table 6. Simulation results

Flows	Type	Requested bandwidth (kb/s)	Measured bandwidth (kb/s)	Routes number	Number of route changes
f1	QoS	120	118	9	11
f2	QoS	140	122	47	51
f3	BE	50	16	7	8
f4	BE	100	10	12	17
f5	BE	80	13	13	15
f6	BE	80	24	3	3
f7	BE	80	18	7	10
f8	BE	50	4	14	13
f9	BE	80	30	2	1
f10	BE	50	9	12	14

Now, we consider the same previous scenario without QoS support (native OLSR). In this case, the two QoS flows respectively obtain 94 kb/s and 70 kb/s (Figure 23).

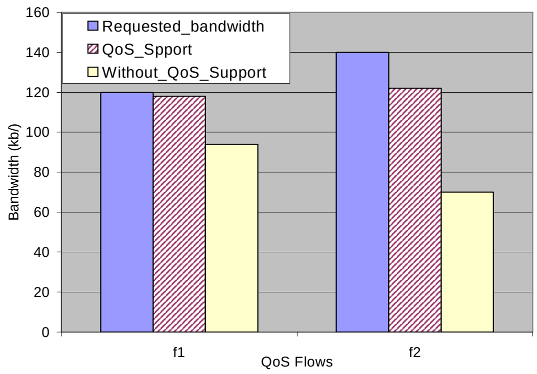

Figure 23. Measured average bandwidth for QoS flows in the two cases: in presence and in absence of QoS support

According to the simulation results, we notice that, with the QoS support, QoS flows obtain more bandwidth. For example, in the scenario above, the first QoS flow *f*1 received a bandwidth (118/ kb/s) close to its required bandwidth (120 kb/s). The second QoS flow *f*2, has received 122 kb/s whereas it requested 140 kb/s. If QoS flows did not obtain the exact requested bandwidth, it is because of the random mobility of the nodes *i.e.*, the random mobility of nodes induces a link failure and consequently a loss of packets. Let us note for example that flow *f*2 changed its route 51 times. A change of route can be due to the one of the three reasons below:

- a link of the current route becomes invalid,
- the current route does not satisfy the required QoS,
- a shorter route satisfying the required bandwidth is found.

Without QoS support, QoS flows see their QoS being degraded. In our example, QoS flows *f*1 and *f*2 obtained only 94 kb/s and 70 kb/s respectively. That is due, on one hand, to the mobility of nodes, and on the other hand to the interferences induced by BE flows which are not taken into account.

4. Conclusion

In this paper, we have proposed a QoS support for mobile ad hoc networks, based on the OLSR routing protocol and the CBQ scheduling. This QoS support takes into account radio interferences and is based on six QoS components: QoS MAC, QoS model, admission control, QoS routing, QoS signaling and scheduling.

User flows are classified according to three types:

- QoS flows with delay constraints
- QoS flows with bandwidth constraints
- BE flows with no specific QoS requirements.

To schedule these three user flow types, we have combined the use of two schedulers which are WCBQ and Priority Queueing (PQ). The CTRL_Queue, dedicated to control traffic, has the highest priority. The Delay_DATA_Queue, dedicated to QoS flows with delay requirements, has lower priority. The Bandwidth_DATA_Queues are dedicated to QoS flows with bandwidth requirement. Best-Effort_DATA_Queues are reserved to BE flows. Bandwidth_DATA_Queues and Best-Effort_DATA_Queues are managed according to CBQ where the weight associated with each flow depends on its bandwidth request. The Delay_Data_Queue is scheduled according to EDF where the local deadline of a packet on a visited node is computed from (*i*) the end-to-end deadline of the flow this packet belongs to, (*ii*) the time already spent by this packet in the network, (*iii*) the number of hops remaining to the destination. This EDF scheduling increases the rate of packets meeting their end-to-end deadline.

We have shown by means of NS2 simulations that this solution provides a fair sharing of bandwidth for best effort flows and ensures route stability for QoS flows. As a consequence, these flows have shorter delays and jitters. As QoS flows are allowed to requisition the bandwidth used by BE flows, they use the shortest route providing the requested QoS. We have also, pointed out the benefits brought by this solution with regard to native OLSR. The overhead of this solution is kept reasonable. Finally, we have shown that our solution supports node mobility.

5. 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] L. Boukhalfa, P. Minet, S. Midonnet, “QoS support in a MANET based on OLSR and CBQ”, *IEEE International Conference on Networking, ICN'07*, Sainte-Luce, Martinique, April 2007.

[4] 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.

[5] S. Floyd and V. Jacobson. "Link-sharing and Resource Management Models for Packet Networks", *IEE/ACM Transactions on Networking*, August 1995, pp. 365-386.

[6] T. Clausen, P. Jacquet , "Optimized Link State Routing Protocol (OLSR) ", IETF RFC 3626, Project Hipercom, INRIA, October 2003.

[7] J. Moy , "OSPF Version 2", IETF RFC 2328, April 1998.

[8] The VINT Project. The network simulator - ns-2 v.2.1b7a. http://www.isi.edu/nsnam/ns/, November 2000.

[9] L. Georgiadis, P. Jacquet, B. Mans, "Bandwidth Reservation in Multihop Wireless Networks: Complexity and Mechanisms", *24th International Conference on Distributed Computing Systems Workshops W6: WWAN (ICDCSW'04)*, 2004, pp. 762-767.

[10] R. Gupta, Z. Jia, T. Tung, J. Walrand, "Interference-aware QoS Routing (IQRouting) for Ad-Hoc Networks", *Globecom*, St. Louis, MO, November 2006.

[11] A. Vers, A. Campbell, M. Barry, L. Sun, "Supporting Service Differentiation in Wireless Packet Networks using Distributed Control", *IEEE Journal on Selected Areas in Communications*, vol.19, N°10, October 2001.

[12] K. Jeffay, D. F. Stanat, C.U. Martel, "On Non-Preemptive Scheduling of Periodic and Sporadic Tasks", *IEEE Real Time Systems Symposium*, December 1991, pp. 129-139.