



Towards an efficient QoS based selection of neighbors in QOLSR

Fadila Khadar, Nathalie Mitton, David Simplot-Ryl

► To cite this version:

Fadila Khadar, Nathalie Mitton, David Simplot-Ryl. Towards an efficient QoS based selection of neighbors in QOLSR. The Third International Workshop on Sensor Networks (SN 2010) in conjunction with IEEE ICDCS 2010, Jun 2010, Genoa, Italy. pp.00. hal-00504964

HAL Id: hal-00504964

<https://hal.science/hal-00504964>

Submitted on 8 Jun 2011

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.

Towards an efficient QoS based selection of neighbors in QOLSR

Fadila Khadar, Nathalie Mitton, David Simplot-Ryl

INRIA Lille - Nord Europe, Univ. Lille 1, CNRS

first-name.surname@inria.fr

Abstract—OLSR is a well-known proactive protocol for wireless networks. Although very efficient by many points, it suffers from the drawbacks of not taking into account QoS metrics such as delay or bandwidth. To overcome this pitfall, some QOLSR (QoS OLSR) solutions have been designed. Nevertheless, they still provide weak performance regarding QoS metrics. In this paper, we introduce a novel and simple neighbor selection to allow routing with OLSR along quasi-optimal QoS paths. Simulation results show that our solution outperforms solutions from the literature by providing efficient QoS paths (less than 2% overhead from optimal paths) by selecting a very small set of neighbors to route packets.

I. INTRODUCTION

Wireless sensor networks (WSNs), *i.e.*, low capabilities devices networks which rely on no infrastructure, present an interesting challenge when it comes to routing. Indeed, each device can act as a router to relay packets of other nodes in the network. The choice of the next relay to be used can be based on several criteria. The most common criterion is the hop distance, which aims at minimizing the number of links of the chosen path. Nevertheless, this approach does not take into account the quality of the link. In a WSN, this quality may be the bandwidth associated to the link, the delay, the reliability of the link, It could also be the level of battery left or the energy consumption associated to this link, which would results in energy-efficient paths.

Routing protocols are usually divided into reactive and proactive approaches. Reactive protocols will typically build a route when needed whereas proactive protocols choose to maintain routing tables in which are stored all routes to known destination. In this article, we focus on a well-known proactive protocol for wireless ad hoc networks, OLSR. The Optimized Link State Routing Protocol (OLSR)

is a widely spread pro-active protocol based on the link state routing protocol designed for MANETs. OLSR performs hop-by-hop routing: each node maintains a routing table that contains the next hop to be used for reaching a given destination. OLSR's optimization over the link state routing protocol is based on the use of Multi-Point Relays (MPRs) and Advertised Neighbor Sets (ANS). MPRs are subsets of nodes that are used for broadcasting topological information in the network while ANS are used for routing packets. They are relative to a node. In the original OLSR protocol, both sets are identical and we only refer to the MPR set. Each node selects a subset of its neighbors as MPRs that should cover its two-hop neighborhood. A node wanting to send a packet sends it to one of its MPRs which will relay it to one of its MPRs and so on until the packet reaches its final destination. OLSR uses control traffic to maintain routes. Topology Control (TC) messages are broadcast through the network using the MPRs. They contain identifier of nodes by which the node has been selected as MPRs (its *MPR selectors*). The heuristic used for the MPRs selection tries to minimize the number of MPRs. It assumes all links are equals, *i.e.*, no link is better than an other. When considering Quality of Service (QoS), this can be a major drawback as these links may not fulfill the requirements of the application like delay or bandwidth constraints. To overcome this problem, several QOLSR (QoS OLSR) have been designed. Nevertheless, they still do not offer consequent QoS increase.

In this paper, we focus on routing in OLSR with QoS constraints. We present a novel ANS selection mechanism that takes into account QoS metrics of any kind, either additive like the delay (delay of a path is the sum of the delay of every hop) or

concave like bandwidth (bandwidth of a path is the minimum bandwidth on links). The choice of bandwidth and delay are for illustration purposes only. The use of other metrics, such as level of battery left, would not modify our algorithm. The presented subset selection should be used in addition to MPR set to provide QoS compliant routes. We show that our solution gets closer to the optimal route than QOLSR and minimizes the size of the ANS set to be advertised in TC messages. In our solution - first node on best path based QANS selection (FNBP) - a node first computes the QoS-weighted shortest path (based on its local view of the graph) towards every of its one-hop and two-hop neighbors. For each of its one-hop and two-hop neighbors, it selects as MPR and advertises the first node on the best paths. Simulation results show that our solution outperforms solutions from the literature by providing efficient QoS paths (less than 2% overhead from optimal paths) by selecting a very small set of neighbors to route packets.

This paper is organized as follows. We first review the heuristics used in OLSR and QOLSR in Section II. We then present our solution for providing QoS in OLSR in Section III. We evaluate this solution in Section IV by comparing its performances to the ones of related works before concluding in Section V.

II. RELATED WORK

OLSR is a proactive routing protocol for wireless network defined in the RFC 3626 [4]. OLSR allows to decrease the overload by reducing the number of nodes involved in a broadcast process through the use of Multipoint relays (MPRs) [9]. They are in charge of disseminating topological information and limiting flooding.

In OLSR, the MPR selection is a two-phase process done according to the coverage of two-hop neighbors by nodes. Each node begins with an empty MPR set. It first selects its one-hop neighbors that are the only to cover some two-hop neighbors as they will have to be in the MPR set anyway. It then picks up the one hop neighbor that covers a maximum number neighbors that are not covered yet. This step stops when all 2-hop neighbors are covered. This heuristic does not take into account the quality of links such as bandwidth, delay, jitter

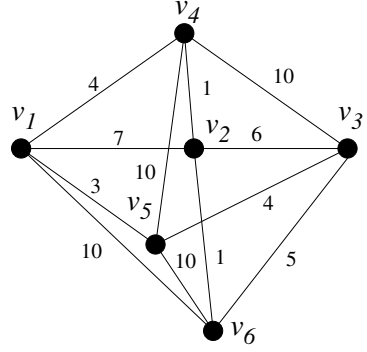


Fig. 1. Example of bad behavior of QOLSR heuristic with the bandwidth metric. Only nodes v_2 and v_5 are selected as MPRs (by v_1, v_3 and by v_6, v_4 respectively). The widest path ($v_1 v_6 v_5 v_4 v_3$, bandwidth of 10) between v_1 and v_3 will not be used by QOLSR.

or stability. The computation of these metrics is out of the scope of this paper, but can be done as proposed in [8].

A QoS version of OLSR, QOLSR [1], proposes two versions of this heuristic to overcome this problem. The first step remains the same, each node first selects the one hop neighbors that have to be in the set as they are the only one to cover some 2-hop neighbors. The two proposed heuristics differ in the second step. The first one, MPR-1, just chooses the node with the highest bandwidth or smallest delay in case of a tie. The second one, MPR-2, does not consider the number of covered 2-hop neighbors but the bandwidth or delay when choosing the next node to add to its MPR set. One drawback in this approach is that, like in the original OLSR, it does not allow to choose a path longer than two hops in order to maintain shortest paths in terms of number of hops. Consequently, the sets of nodes selected by these heuristics do not differ much from the one selected in the original OLSR heuristic. Indeed, it has been shown in [3] that 75% of the MPRs are selected during the first phase as they are the only ones to cover some 2-hop neighbors.

Another feature of the heuristic used in QOLSR is illustrated in Figure 1 in which a network of six nodes is depicted. Weight on each link indicates the bandwidth it can provide. When considering this network, only node v_2 and v_5 are selected as MPRs. v_2 's MPR selectors are v_4 and v_6 while v_5 is selected

by v_1 and v_3 . Therefore, when v_1 wants to reach v_3 , it uses v_2 as relay. The bandwidth associated to this path is 6. Thus, the optimal path $v_1v_6v_5v_4v_3$, which associated bandwidth is 10, will not be used, unlike claimed in [1].

In [7] the authors propose to use different sets of nodes for flooding and routing. The former one, used for broadcasting of TC messages, is the classical MPR set. The latter set, called the QoS Advertised Neighbor Set (QANS) is used for routing packets. It takes into account QoS metrics. Its selection is based on topology filtering: the node first performs a reduction of the graph represented by the neighborhood it is aware of. The graph reduction algorithm used is the relative neighborhood graph [10] with bandwidth or delay as a weight function. A node is in the QANS set if it maximizes (minimizes respectively) bandwidth (delay respectively) to a 2-hop neighbor in the reduced graph. As a result, and unlike QOLSR, a two-hop path can be used for reaching a one-hop neighbor if it offers better QoS. Yet, this approach still suffers from the same drawback as QOLSR, as it does not allow to select a path with a number of hops greater than 2.

Although this approach reduces the size of the selected set, it suffers from the drawback of selecting several paths with the best QoS. As they will all be selected as advertised neighbors, the cardinality of the set is still quite higher than the one of the optimal solution.

We preserve the mechanisms introduced in OLSR but we use the same approach as in [7], *i.e.*, we build both MPR and QANS sets. The MPR set is computed using the heuristic proposed in the original OLSR. Our goal and main focus is to reduce the size of the QANS set. Our approach allows a node to select a path which length in number of hops is greater than 2 for reaching a two-hop neighbor and reduces the number of advertised neighbors while selecting paths with greater bandwidth or smaller delay. Furthermore, we allow a node to select an advertised neighbor for reaching a 1-hop neighbor if this selected node offers better bandwidth.

As mentioned earlier, we do not focus on the computation of QoS metric to be used. The reader can refer to [2] for example of how these metrics (bandwidth and delay) can be computed. Other metrics can be used such as the one proposed in [6]

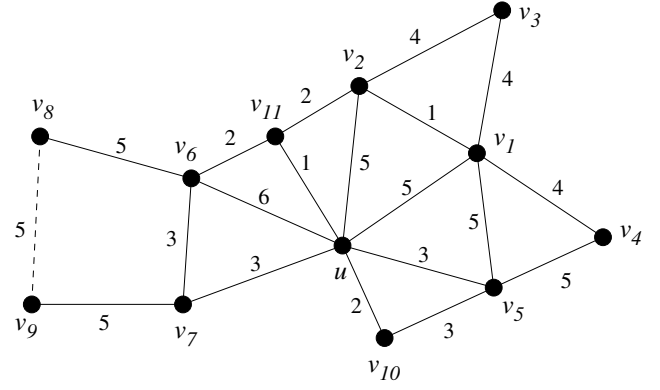


Fig. 2. Example of network. The metric used is bandwidth. Only the two-hop neighborhood of node u is depicted. Dashed links represent links u is not aware of.

in which residual energy (energy consumed when sending a message) is considered.

III. QANS SELECTION ALGORITHM

In this section we detail the algorithm used for selecting the advertised neighbor set. This set of nodes will be used for routing and is advertised in the TC messages. We describe our algorithm with two different kinds of metrics: a concave metric (bandwidth) and an additive metric (delay). These results can be applied for other metrics, like jitter or packet loss metrics which are also additive metrics. Another concave metric could be for example the number of buffers available at each node along a path.

A. Notations

We represent a wireless network by a graph $G = (V, E)$ in which V is the set of nodes belonging to the network and $E \subseteq V^2$ the communication links between these nodes. (u, v) is in E if and only if the distance is shorter than the communication radius R , *e.g.* if and only if $|uv| \leq R$ where $|uv|$ is the Euclidean distance between u and v . We consider that all nodes use the same communication radius and that links are bi-directional.

Each ANS is relative to a node. We thus note $ANS(u)$ the QoS-ANS of node u . We note $N(u)$ the set of neighbors of node u , *i.e.* the set of nodes with which u can communicate directly and $N_2(u)$ the set of its two-hop neighbors. $N(u)$ and $N_2(u)$:

$ANS(u)$	QOS-ANS of u
$G_u = (V_u, E_u)$	graph viewed by u
$\mathcal{BW}(u, v)$	bandwidth associated to link (u, v)
$\mathcal{P}_{\mathcal{BW}}(u, v)$	set of paths from u to v with maximum bandwidth
$f\mathcal{P}_{\mathcal{BW}}(u, v)$	set of first nodes of best paths
$\widetilde{\mathcal{BW}}(u, v)$	maximum bandwidth between u and v
$\prec_{BW}^u$	ordering operator on bandwidth
$\mathcal{D}(u, v)$	delay associated to link (u, v)
$\mathcal{P}_{\mathcal{D}}(u, v)$	set of paths from u to v with minimum delay
$f\mathcal{P}_{\mathcal{D}}(u, v)$	set of first nodes of best paths
$\widetilde{\mathcal{D}}(u, v)$	maximum bandwidth between u and v
$\prec_D^u$	ordering operator on delay

Fig. 3. Notations used in this article for bandwidth and delay metrics.

$$v \in N(u) \Leftrightarrow (uv) \in E$$

$$v \in N_2(u) \Leftrightarrow$$

$$(\{v\} \mid \exists w \in N(u) \cap N(v) \wedge v \notin N(u) \cup \{u\}).$$

A node u has only a partial view of the graph. This view is noted $G_u = (V_u, E_u)$ with:

$$\begin{aligned} V_u &= \{u\} \cup N(u) \cup N_2(u) \\ E_u &= \{(vw) \mid (v \in N(u) \wedge w \in V_u)\} \end{aligned}$$

The bandwidth and the delay on a link between nodes u and v are noted $\mathcal{BW}(u, v)$ and $\mathcal{D}(u, v)$ respectively.

Let $p(x_0, x_n)$ be a path between nodes x_0 and x_n in G :

$$p(x_0, x_n) = x_0 x_1 \dots x_i x_{i+1} \dots x_n,$$

$$\forall i \ x_i \in V, (x_i, x_{i+1}) \in E.$$

The bandwidth and delay of such a path p are noted $\mathcal{BW}(p)$ and $\mathcal{D}(p)$ resp. and are computed as follows:

$$\begin{aligned} \mathcal{BW}(p) &= \min_{0 \leq i < n-1} \mathcal{BW}(x_i, x_{i+1}), \\ \mathcal{D}(p) &= \sum_{i=0}^{n-1} \mathcal{D}(x_i, x_{i+1}). \end{aligned}$$

Let $\mathcal{P}_{\mathcal{BW}}(u, v)$ be the set of paths from u to v with maximum bandwidth and $\widetilde{\mathcal{BW}}(u, v)$ the value of this bandwidth:

$$\widetilde{\mathcal{BW}}(u, v) = \max_{p \in \mathcal{P}_{\mathcal{BW}}(u, v)} \{\mathcal{BW}(p)\}.$$

We note $f\mathcal{P}_{\mathcal{BW}}(u, v)$ the set of nodes w such that w is the first node on path p for every p in $\mathcal{P}_{\mathcal{BW}}(u, v)$. For instance, on Fig. 2, we have $\mathcal{P}_{\mathcal{BW}}(u, v_3) = \{uv_2v_3, uv_1v_3\}$ of bandwidth value $\widetilde{\mathcal{BW}}(u, v_3) = 4$ and $f\mathcal{P}_{\mathcal{BW}}(u, v_3) = \{v_2, v_1\}$. Similarly, $\mathcal{P}_{\mathcal{D}}(u, v)$ designates the set of paths with minimum delay between u and v , $\widetilde{\mathcal{D}}(u, v)$ the value of this delay and $f\mathcal{P}_{\mathcal{D}}(u, v)$ the set of first nodes on best paths between u and v .

For each node u we define the binary total ordering operator $\prec_{BW}^u$ as follows:

$$w \prec_{BW}^u v, \forall u, v, w \in V, v, w \in N(u)$$

$$\text{iff } \mathcal{BW}(u, w) > \mathcal{BW}(u, v)$$

$$\text{or } \mathcal{BW}(u, w) = \mathcal{BW}(u, v) \wedge (id_v < id_w).$$

In short, $w \prec_{BW}^u v$ if nodes v and w are both neighbors of node u and either that link between u and v offers a better bandwidth than the link between nodes u and w or that both links uv and uw offer the same bandwidth and that v has a smaller identifier than w . For instance, on Fig. 2, we have $v_5 \prec_{BW}^u v_1$ as $\mathcal{BW}(u, v_5) < \mathcal{BW}(u, v_1)$. The bandwidth between u and v_1 being the same as between u and v_2 , $v_1 \prec_{BW}^u v_2$ because v_1 has a smaller identifier than v_2 . Let $\max_{\prec_{BW}^u}$ denote the maximum function associated to this total order. Likewise, we specify the binary total order $w \prec_D^u v$ such that $w \prec_D^u v, \forall u, v, w \in V, v, w \in N(u)$ if and only of $\mathcal{D}(u, w) < \mathcal{D}(u, v)$ or $\mathcal{D}(u, w) = \mathcal{D}(u, v) \wedge (id_v < id_w)$. $\min_{\prec_D^u}$ denotes the minimum function associated to this total order.

B. Algorithm Description

As already mentioned, in the original OLSR proposition [4], MPRs are used for two purposes: broadcast and diffusion of TC messages for establishing routing tables. Original MPRs are selected in such a way that shortest paths in terms of number of hops are preserved. Although this is the most adequate strategy for broadcasting, this is not necessarily the case for routing with QoS constraints [7]. We adopt the approach proposed in [7] in which ANS and MPR are distinct sets. The underneath idea of this proposition is that a direct link with a node may not be optimal in term of QoS (bandwidth or delay for example). For getting

better performances, we allow a node to choose a k -hop path ($k \geq 2$) instead of a direct link if this path offers better performances. For example in Figure 2, u must be able to choose path $uv_1v_5v_4$ to reach v_4 , achieving a bandwidth of 5, rather than the direct link of bandwidth 3.

Providing optimized paths in terms of QoS between any two nodes in a wireless ad hoc network can be achieved only through a centralized mechanism. As Figure 2 shows, this can never be accomplished through the use of localized protocols unlike claimed in [1]. A localized algorithm should be able to take decisions upon knowledge about its k -hop neighborhood, k being constant. In our example, as $k = 2$, node u is not aware of link (v_8v_9) . It will thus choose path uv_7v_9 with bandwidth of 3 to reach v_9 while path $uv_6v_8v_9$ with a bandwidth of 5 exists. Yet, our algorithm has the following goals: providing in a localized way, the best possible paths in term of QoS between two 2-neighbor nodes by minimizing the number of ANS selected. We assume that every node u has only knowledge of its neighboring graph G_u as in the original QOLSR protocol [4]. This can be achieved by piggybacking neighborhood table in Hello messages.

FNBP runs in two steps. The first step (from line 1 to line 7 in Algorithm 1 for bandwidth and in Algorithm 2 for delay) is dedicated to the selection of ANS to reach 1-hop neighbors with the best QoS. An ANS is selected by node u to reach a 1-hop neighbor v if and only if the direct link (u, v) is not the best path in terms of QoS requirements to reach v from u (condition line 2 of Alg. 1 and 2). The second step then focuses on the selection of ANS for 2-hop neighbors. Both steps proceed similarly. In the first step, for each of its 1-neighbors v , node u computes the best QoS paths between itself and the selected neighbor v in G_u . It retrieves the set $f\mathcal{P}_{BW}(u, v)$ (or $f\mathcal{P}_D(u, v)$) from this computation. If $v \in f\mathcal{P}_{BW}(u, v)$ (or $v \in f\mathcal{P}_D(u, v)$), that means that the direct link (uv) is optimized with regards to the QoS and thus u does not need to select an additional ANS to cover node v since it will already receive the message through an optimal link. In Fig. 2, node u will therefore not select another ANS for reaching node v_7 as the direct link (uv_7) provides the best bandwidth.

If the direct link does not offer the best band-

width, u compares $f\mathcal{P}_{BW}(u, v)$ set to the current ANS set it already has. If u has already selected an ANS node w on one of the optimal paths ($w \in f\mathcal{P}_{BW}(u, v) \cap ANS(u)$ or $w \in f\mathcal{P}_D(u, v) \cap ANS(u)$), there is no use to select an additional node as ANS to cover node v since it will already be covered through a path via w . Back to the example in Figure 2, assume u first selects v_1 for reaching v_5 , when choosing a ANS for reaching node v_{10} , it will choose v_1 over v_5 as it is already in its ANS. But if no such node exists ($f\mathcal{P}_{BW}(u, v) \cap ANS(u) = \emptyset$ or $f\mathcal{P}_D(u, v) \cap ANS(u) = \emptyset$), node u selects as ANS the node w in $f\mathcal{P}_{BW}(u, v)$ (or $f\mathcal{P}_D(u, v)$) such that the direct link uw presents the best QoS value. In case of ties, the smallest id is preferred, *i.e.* u selects the node w such that $\max_{\prec_{BW}}(f\mathcal{P}_{BW}(u, v))$ (or $\min_{\prec_D}(f\mathcal{P}_{BW}(u, v))$). In our example, u will choose v_6 instead of v_2 for covering v_{11} as the link (u, v_6) offers a better bandwidth.

The second step is similar to the first one except that u now considers its 2-hop neighbors v (from lines 8 to 17 of Alg. 1 and 2). The first phase consists in the computation of the QoS-weighted shortest paths from u to v in G_u and in the extraction of the set $f\mathcal{P}_{BW}(u, v)$ (or $f\mathcal{P}_D(u, v)$). From it, u checks whether it has already selected as a ANS one of the nodes from this set (line 9 of Alg. 1 and 2). If so, like for the first step of the ANS selection, there is no use to select an additional node. This is illustrated in Figure 2. No additional node will be selected for reaching v_3 as v_1 is already in $ANS(u)$ and belongs to $f\mathcal{P}_{BW}(u, v)$. If not ($f\mathcal{P}_{BW}(u, v) \cap ANS(u) = \emptyset$ or $f\mathcal{P}_D(u, v) \cap ANS(u) = \emptyset$), node u selects as ANS the node w in $f\mathcal{P}_{BW}(u, v)$ (or $f\mathcal{P}_D(u, v)$) such that the direct link (u, w) presents the best QoS value (lines 9 and 10 of Alg. 1 and 2). In case of a tie, the smallest id is preferred, *i.e.* u selects the node w such that $\max_{\prec_{BW}}(f\mathcal{P}_{BW}(u, v))$ (or $\min_{\prec_D}(f\mathcal{P}_{BW}(u, v))$).

Nevertheless, this is not enough to guarantee the message delivery. Indeed, if we have a look on Figure 4, node B will select A for reaching E (link (BA) provides a better bandwidth than link (BC) and will have to be selected anyway to cover D). Similarly, A will select B to reach E as it is already in the ANS. This situation creates a loop and E becomes unreachable since node D is the only access to E : D has been selected by no node.

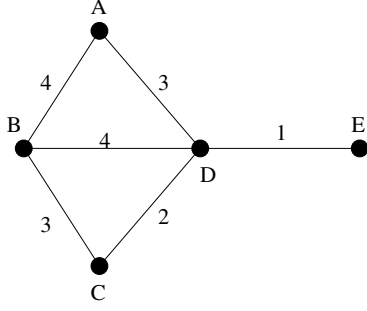


Fig. 4. Example of bad behavior when the last link is a limiting QoS link. A and B select each other for reaching E, creating a loop.

This occurs when the last link is a limiting QoS link.

To overcome this drawback, the following condition is added. If u has a smaller id than every node in QoS-weighted shortest paths it has to select as a ANS a node w such that the path uwv exists. In our example (Fig. 4), A will have to select D to reach E. An example of the subset selected is shown in Fig. 5. The example also includes the selected MPR set and the ANS selected by topology filtering. Algorithms 1 and 2 describe the ANS selection we propose for bandwidth and delay respectively.

IV. EVALUATION

A. Simulation settings

We evaluate the performance of FNBP (first node on best path based QANS selection) and compare it to the original QOLSR (with the MPR-2 heuristic [1]) and the topology filtering proposed in [7]. In order to focus on the routing layer feature and eliminate the effect of the MAC layer on our results, we use our own C simulator that assumes an ideal MAC layer, i.e. no interferences and no packet collisions. Nodes are deployed in a 1000×1000 square using a Poisson Point Process (node positions are independent) with different node degree¹ δ . The communication radius R is set to 100. The results are the average of 100 runs in which a source (noted u) and a destination (v) are randomly chosen. Weights (QoS values) on links are uniformly drawn at random in a fixed interval. Each approach is run

¹In such a Poisson Point Process, the total number of nodes is probabilistic and is obtained from a Poisson Law of intensity λ with $\lambda = \frac{\delta}{\pi R^2}$.

```

Input:  $u \in V, N(u), N_2(u)$ 
Output:  $ANS(u)$ 
// Step 1: selection of ANS for
// 1-neighbors
1 forall  $v \in N(u)$  do
2   if  $f\mathcal{P}_{BW}(u, v) \cap ANS(u) = \emptyset$  then
3     if  $\max_{\prec}(f\mathcal{P}_{BW}(u, v)) \neq v$  then
4        $ANS(u) \leftarrow \max_{\prec_{BW}}(f\mathcal{P}_{BW}(u, v));$ 
5     end
6   end
7 end
// Step 2: selection of ANS for
// 2-neighbors
8 forall  $v \in N_2(u)$  do
9   if  $f\mathcal{P}_{BW}(u, v) \cap ANS(u) = \emptyset$  then
10     $ANS(u) \leftarrow \max_{\prec_{BW}}(f\mathcal{P}_{BW}(u, v));$ 
11   else
12     if  $\{min_{id}(f\mathcal{P}_{BW}(u, v)) > u\} \wedge$ 
13        $\{\exists w \in f\mathcal{P}_{BW}(u, v) \cap N(u)\}$  then
14          $ANS(u) \leftarrow$ 
15            $\max_{\prec_{BW}}(f\mathcal{P}_{BW}(u, v) \cap N(u));$ 
16       end
17   end
18 end

```

Algorithm 1: ANS Selection at node u using bandwidth as a metric.

on the same topology with the same source and destination.

As already mentioned, the goal of our algorithm is to provide paths with maximum bandwidth (or minimum delay) while minimizing the size of the set of QANS at every node. We first analyze the size of the QANS set. Then, to further evaluate the quality of protocols, we compute their delay or bandwidth overhead using as reference the optimal centralized energy weighted SP (Dijkstra algorithm [5]). We let b (resp. d) and b^* (resp. d^*) be the bandwidth (resp. delay) consumed to route a packet from u to v using any described protocol and the centralized protocol, respectively. We define the bandwidth overhead (resp. delay overhead) as the ratio $\frac{b^*-b}{b^*}$ (resp. $\frac{d-d^*}{d^*}$). Yet, the bandwidth overhead represents the bandwidth that should have been used by protocols compared to the optimal solution while the delay overhead represents the delay that should have been saved.

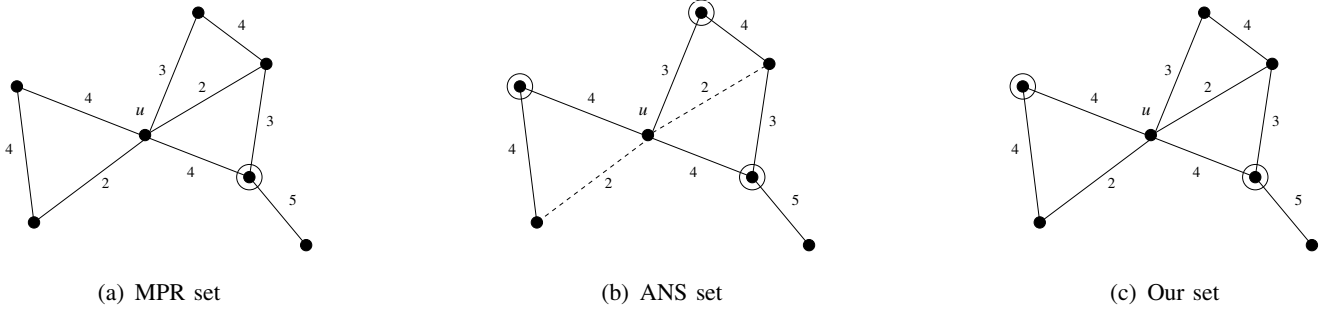


Fig. 5. Example of subset selection. Figure 5(a) represents the set of MPRs of u . The subset selected by the approach of [7] is shown in Figure 5(b). The dashed links represent links removed by topology filtering. The subset selected by our algorithm is presented on Figure 5(c).

```

Input:  $u \in V, N(u), N_2(u)$ 
Output:  $ANS(u)$ 
// step 1: selection of ANS for
// 1-neighbors
1 foreach  $v \in N(u)$  do
2   if  $f\mathcal{P}_D(u, v) \cap ANS(u) = \emptyset$  then
3     if  $\min_{\prec}(f\mathcal{P}_D(u, v)) \neq v$  then
4        $ANS(u) \leftarrow \min_{\prec_D}(f\mathcal{P}_D(u, v));$ 
5     end
6   end
7 end
// Step 2: selection of ANS for
// 2-neighbors
8 forall  $v \in N_2(u)$  do
9   if  $f\mathcal{P}_D(u, v) \cap ANS(u) = \emptyset$  then
10     $ANS(u) \leftarrow \min_{\prec_D}(f\mathcal{P}_D(u, v));$ 
11   else
12     if  $\{min_{id}(f\mathcal{P}_D(u, v)) > u\} \wedge$ 
13        $\{\exists w \in f\mathcal{P}_D(u, v) \cap N(u)\}$  then
14          $ANS(u) \leftarrow$ 
15            $\min_{\prec_D}(f\mathcal{P}_D(u, v) \cap N(u));$ 
16       end
17   end
18 end

```

Algorithm 2: ANS Selection at node u using delay as a metric.

B. Results

a) Size of the Advertised Set: For each approach we compare the size of the set of neighbors that has to be advertised in TC messages. The results are shown in Figures 6 and 7 for bandwidth and delay metrics respectively. Results are similar for both metrics. Our approach presents the best performance followed by the topology-filtering based

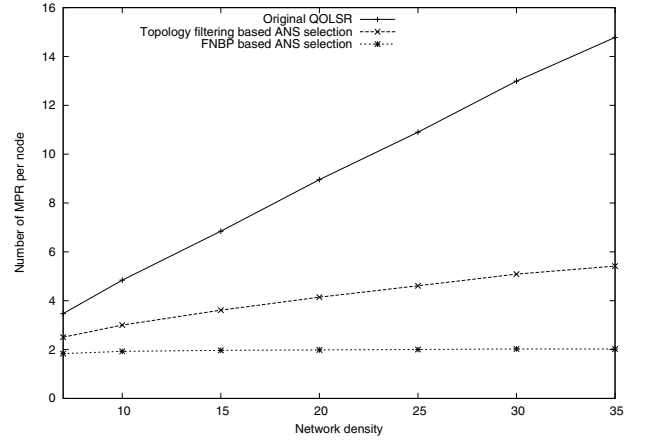


Fig. 6. Size of the set that has to be advertised in TC messages using bandwidth as a metric.

approach and last by the original QOLSR. The original heuristic in OLSR computes a MPR set of cardinality at most $\log n$ times the optimal MPR set, n being the number of nodes in the network [9]. Unlike other approaches, the size of the advertised set of FNBP remains constant even if the network density increases. This is due to the fact that when the network is dense, a new QANS is added if and only if there is no other adequate node already selected. Since the denser the network, the less likely nodes may be isolated, this set thus remains constant. This is not the case for other solutions since they automatically select the first node on every path with maximum bandwidth. Since the number of nodes increases, so the number of paths and thus the number of nodes to be selected.

b) Overhead: The bandwidth overhead of the different solutions is displayed on Figure 8 while

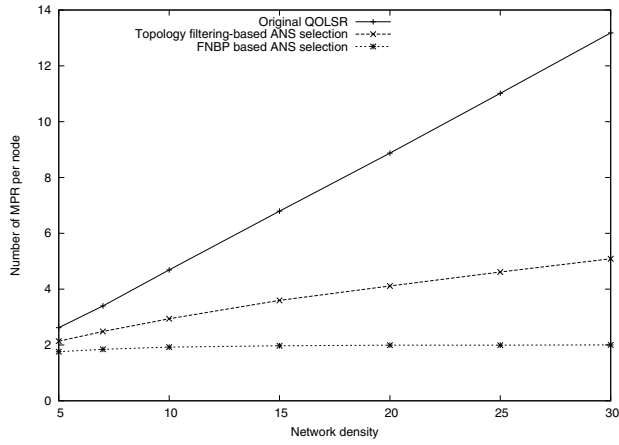


Fig. 7. Size of the advertised set using delay as a metric.

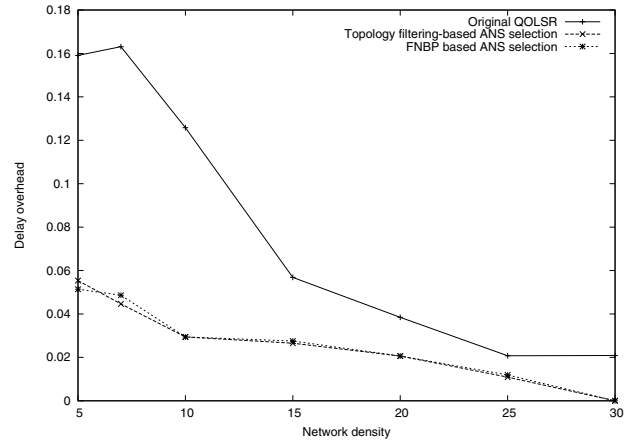


Fig. 9. Delay overhead compared to the optimal solution.

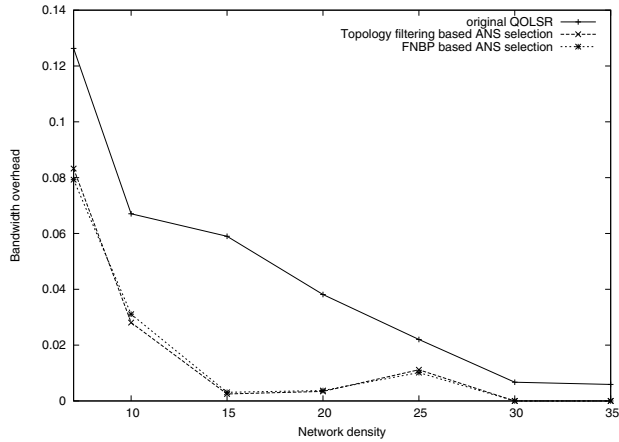


Fig. 8. Bandwidth overhead generated by our solution, QOLSR and the solution proposed in [7] compared to the optimal solution (computed in a centralized way).

Figure 9 shows the delay overhead. Results are similar in both cases. Our solution provides similar overhead that the topology-filtering based approach [7] using both delay or bandwidth metrics, outperforming the original solution. This is due to the fact that in both cases, same paths are followed, maximizing the QoS in every two-hop neighborhood. The overhead tends to decrease as density increases, getting closer to the optimal centralized solution.

V. CONCLUSION

This article presented a heuristic for the selection of Quality Advertised Neighbor Set. This set is meant to be used with the OLSR protocol and

provides quality of service for this protocol. This heuristic may be employed with additive or concave metrics. It reduces the size of the advertised set while offering a low overhead compared to the optimal solution computed in a centralized manner. Future work will focus on multi-criterion metrics for example minimizing energy-consumption while providing good bandwidth.

VI. ACKNOWLEDGEMENTS

This work has been supported by the FP6 WASP² (Wirelessly Accessible Sensor Populations) project.

REFERENCES

- [1] H. Badis and K. A. Agha. QOLSR, qos routing for ad hoc wireless networks using OLSR. *European Transactions on Telecommunications*, 16(5):427–442, 2005.
- [2] H. Badis, A. Munaretto, K. Al Aghal, and G. Pujolle. Optimal path selection in a link state QoS routing protocol. In *IEEE 59th Vehicular Technology Conference. (VTC 2004-Spring)*, volume 5, pages 2570–2574, May 2004.
- [3] A. Busson, N. Mitton, and E. Fleury. An analysis of the MPR selection in OLSR and consequences. In *Mediterranean Ad Hoc Networking Workshop (MedHocNet'05)*, France, June 2005.
- [4] T. Clausen and P. Jacquet. Optimized link state routing protocol (OLSR). RFC 3626, Internet Engineering Task Force, 2003.
- [5] E.W. Dijkstra. Solution of a problem in concurrent programming control. *Communications of the ACM*, 8(9), September 1965.
- [6] S. Mahfoudh. *Energy efficiency in Wireless ad hoc and sensor networks: routing, node activity scheduling and cross-layering*. PhD thesis, Universit   Pierre et Marie Curie, Paris XI, 2010.
- [7] L. Moraru and D. Simplot-Ryl. Qos preserving topology advertising reduction for OLSR routing protocol for mobile ad hoc networks. *3rd Annual Conference on Wireless On demand Network Systems and Services (WONS)*, December 2006.

²<http://www.wasp-project.org/>

- [8] A. Munaretto and M. Fonseca. Routing and quality of service support for mobile ad hoc networks. *Computer Networks*, 51(11):3142 – 3156, 2007.
- [9] A. Qayyum, L. Viennot, and A. Laouiti. Multipoint relaying for flooding broadcast messages in mobile wireless networks. In *35th Hawaii International Conference on System Sciences (HICSS)*, volume 9, page 298, Los Alamitos, CA, USA, January 2002. IEEE Computer Society.
- [10] G. T. Toussaint. The relative neighbourhood graph of a finite planar set. *Pattern Recognition*, 12:261–268, 1980.