

Improved flooding of broadcast messages using extended multipoint relaying

Pere Montolio Aranda^a, Joaquin Garcia-Alfaro^{a,b}, David Megías^a

^a *Universitat Oberta de Catalunya, Estudis d'Informàtica, Multimèdia i Telecomunicació.
Rambla del Poblenou 156, 08018, Barcelona, Spain*

^b *Institut Télécom. Dépt. Logique des Usages, Sciences Sociales et de l'Information.
Télécom Bretagne, Cesson-Sévigné, CS 17607, 35576, France*

E-mails: pma@uoc.edu, joaquin.garcia-alfaro@acm.org, dmegias@uoc.edu

Abstract

A common operation in *wireless ad hoc* networks is the flooding of broadcast messages to establish network topologies and routing tables. The flooding of broadcast messages is, however, a resource consuming process. It might require the retransmission of messages by most network nodes. It is, therefore, very important to optimize this operation. In this paper, we first analyze the *multipoint relaying* (MPR) flooding mechanism used by the *Optimized Link State Routing* (OLSR) protocol to distribute topology control (TC) messages among all the system nodes. We then propose a new flooding method, based on the fusion of two key concepts: distance-enabled *multipoint relaying* and *connected dominating set* (CDS) flooding. We present experimental simulations that show that our approach improves the performance of previous existing proposals.

Key words: Wireless, Ad Hoc Networks, Network Flooding, OLSR, MPR, CDS.

1. Introduction

Wireless ad hoc networks are generally constructed without a preestablished structure. The nodes are in charge of discovering the system topology that will eventually allow them to work together. Two usual mechanisms to share information between the network nodes are: (1) point-to-point, in which a node sends its messages towards a unique destination; and (2) flooding of broadcast messages, in which a node broadcasts a message to all its neighbors, and those neighbors (or part of them) forward, recursively, the original message to each node of the network [12]. The flooding process can be used at several logical levels [2, 3]. For instance, it can be used at the application level (when a node needs to send information to each node of the network with a query or a declaration of data interest); or at the network control level (for the consolidation of the network structure, existence of nodes, or the characteristics and availability of the network links). These network control messages must be sent not only during the network consolidation stage, but also during all the network life (in order to update the nodes with environmental and network changes).

The resource consumption of flooding a network is, in general, very high. It needs several data transmission actions to be executed by high number of nodes in the network. For this reason, the optimization of any flooding mechanism is mandatory. In this sense, the *Optimized Link State Routing* (OLSR) protocol [4] suggests an optimization for the flooding of broadcast network consolidation messages on *ad hoc* networks based on multipoint relaying (MPR) flooding. Afterwards, it is possible to optimize the flooding of the remaining broadcast messages by reorganizing the initial MPR flooding mechanism into

a *Connected Dominating Set* (CDS) structure. In this paper, we analyze these methods and transformations formally and then, we present the design of an alternative variant based on the use of an extended multipoint relaying flooding that improves the performance of the resulting CDS structure highly. We present a series of simulations that show the validity of our approach, and that reveal that our method considerably improves the performance of previous existing proposals. Our proposal is, moreover, compatible with the forthcoming evolutions of the OLSR protocol (referred as OLSRv2 in the literature), by addressing the sketched extended quality criteria, such as the security aspects of the resulting new protocol [9, 10].

Notation — Some basic notations are necessary to describe the proposals and operations we present in this work:

- $\text{dist}(i, j)$: distance, in hops, from node i to node j .
- $N_1(x) = \{y \mid \text{dist}(x, y) = 1\}$: neighbors of a node, *i.e.*, adjacent nodes.
- $N_1^*(x) = N_1(x) \cup \{x\}$: neighbors of a node x , including the node itself.
- $N_1^*(X) = \bigcup_{x \in X} N_1^*(x)$, neighbors of a set, including the set itself.
- $N_d^*(X)$, neighbors of a set at a distance in hops lower or equal to d :
$$N_d^*(X) = \begin{cases} X, & \text{if } d = 0, \\ N_{d-1}^*(N_1^*(X)), & \text{if } d > 0. \end{cases}$$
- $N_d(X) = N_d^*(X) \setminus N_{d-1}^*(X)$, neighbors of a set at a distance, in hops, strictly equal to d .

Paper organization — Section 2 presents the MPR selection problem and describes some appropriate implementation methods. Section 3 analyzes the flooding process of the OLSR protocol and points out some drawbacks and early improvements. Section 4 proposes our alternative variant. Section 5 presents experimental evaluations and shows the performance results obtained with our proposal. Section 6 closes the paper.

2. Multipoint relay (MPR) selection

The multipoint relay (MPR) selection problem appears as an early stage of the OLSR routing protocol. The original problem formulation is as follows: given a network node x , find a minimal subset of nodes, $\text{MPR}(x)$, included in their immediate neighborhood with the characteristic that any node at a two-hop distance from x is a neighbor of, at least, a node in the MPR:

$$\begin{aligned} \text{MPR}(x) &\subset N_1(x), \\ N_2(x) &\subset N_1(\text{MPR}(x)). \end{aligned}$$

It is a quality objective that $\text{MPR}(x)$ must have minimal number of elements. It can be demonstrated, moreover, that finding the minimal solution is an NP-complex problem [5], since finding a solution is exponential in execution time relative to the number of nodes in $N_1(x)$.

The original MPR selection problem can be generalized with the following formulation: given a set C of candidate nodes and a set T of target nodes, where all target nodes are neighbors of at least one candidate node ($T \subset N_1(C)$), find a subset $\text{MPR}(C, T)$ included in C with minimal size and the characteristic that any node of T is a neighbor of (*i.e.*, it is covered by) a node in $\text{MPR}(C, T)$.

$$\begin{aligned} \text{Precondition: } &T \subset N_1(C), \\ &\text{MPR}(C, T) \subset C, \\ &T \subset N_1(\text{MPR}(C, T)). \end{aligned}$$

The quality criteria is to minimize the number of elements in $\text{MPR}(C, T)$. From the formulation, it is straightforward to note that the original problem described in the OLSR standard is a particular instance of this one, when $C = N_1(x)$ and $T = N_2(x)$. The same conditions can be rewritten as follows:

$$\begin{aligned} \text{Precondition: } &T \subset N_1(C), \\ &\text{MPR}(C, T) \subset C, \\ &\forall y, y \in T \rightarrow |N_1(y) \cap \text{MPR}(C, T)| \geq 1. \end{aligned}$$

2.1. MPR selection with extra coverage

In some network deployments, it is mandatory that any node in T must be neighbor of more than one node in the MPR set. This minimum number of nodes is refereed as the MPR coverage (denoted here as m_{cvg}) in the OLSR protocol (cf. RFC 3626 [4], Section 16). The previous problem formulation was the one for which the m_{cvg} factor is equal to one. The generalized conditions for any given m_{cvg} are the following:

$$\begin{aligned} \text{MPR}(C, T, m_{\text{cvg}}) &\subset C, \\ \forall y, y \in T &\rightarrow |N_1(y) \cap \text{MPR}(C, T, m_{\text{cvg}})| \geq m_{\text{cvg}}. \end{aligned}$$

It might not always be possible to fulfill these conditions if one node in T has not enough (*i.e.*, m_{cvg}) valid candidates in C . If that happens, the second condition must be relaxed to include all available neighbors. The resulting condition is, hence, rewritten as follows:

$$\begin{aligned} \text{MPR}(C, T, m_{\text{cvg}}) &\subset C, \\ \forall y, y \in T &\rightarrow |N_1(y) \cap \text{MPR}(C, T, m_{\text{cvg}})| \geq m_{\text{cvg}} \\ &\vee N_1(y) \cap C \subset \text{MPR}(C, T, m_{\text{cvg}}). \end{aligned}$$

2.2. MPR selection as the solution of a restrictions problem

The MPR problem without extra coverage can be seen as the solution of a boolean satisfiability problem. The variables of this problem are as many booleans as elements in C . The *false* value means that this element of C is not included in the set $\text{MPR}(C, T)$, whereas the *true* value means that this element of C is included in $\text{MPR}(C, T)$. These variables are denoted as c_0, c_1, c_2 , and so on. Each node in T adds a new restriction to the set of possible solutions of the problem. This rule is an *or* condition for the variables related to their neighbors in C . The objective is to set to *true* a minimal number of variables c_x , such that all rules are *true*.

Assume the network depicted in Figure 1, where gray nodes represent the set C and white nodes the set T .

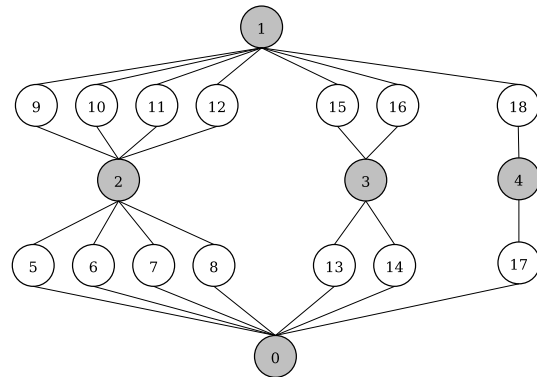


Figure 1: MPR problem example.

We can observe that node 5 has two neighbors in C , nodes 0 and 2. Thus, it generates the restriction $c_0 \vee c_2$. The remaining nodes in T generate the following restrictions:

$$\begin{aligned} t_5 &: c_0 \vee c_2, \\ t_6 &: c_0 \vee c_2, \\ t_7 &: c_0 \vee c_2, \\ t_8 &: c_0 \vee c_2, \\ t_9 &: c_1 \vee c_2, \\ t_{10} &: c_1 \vee c_2, \\ t_{11} &: c_1 \vee c_2, \\ t_{12} &: c_1 \vee c_2, \\ t_{13} &: c_0 \vee c_3, \\ &\dots \end{aligned}$$

This problem has one optimal solution $c_0 \wedge c_1$. The combination $c_2 \wedge c_3 \wedge c_4$ is also a solution although it is not minimal.

2.3. Solution of a restrictions problem with extra coverage

When the MPR coverage parameter (m_{cvg} for short) is greater than one (cf. RFC 3626 [4], Section 16), the problem must be reformulated. Each node in C is now associated to an integer variable in $0, 1$. Each node in T adds one restriction to the set of possible solutions of the form *addition of their neighbors greater or equal than the restriction*. The previous example, now with $m_{cvg} = 2$, i.e.,

$$\begin{aligned} t_5 : c_0 + c_2 &\geq 2 \vee c_0 = c_2 = 1, \\ t_6 : c_0 + c_2 &\geq 2 \vee c_0 = c_2 = 1, \\ \dots \\ t_9 : c_1 + c_2 &\geq 2 \vee c_1 = c_2 = 1, \\ \dots \end{aligned}$$

has only one possible solution $c_1 = c_2 = c_3 = c_4 = c_5 = 1$. Let us observe, from this example, that:

- Several nodes in C are mandatory in the MPR set. If one target node has a number of neighbors in C equal or smaller than the requested coverage, these neighbors are mandatory in the MPR. Looking at the example depicted in Figure 1, and if $m_{cvg} = 2$, it is obvious that node 5 forces c_0 and c_2 to be included in the MPR subset.

This situation is very likely to happen. Indeed, in most of the experimental simulations that we present in Section 5, this kind of *poorly covered* nodes are, in most cases, almost the 90% of the total nodes in C .

- Several nodes in T generate the same restriction, such as nodes 5, 6, 7 and 8 depicted in Figure 1. When a specific strategy, like the greedy strategy used in the OLSR protocol [5], needs to assign a weight to each candidate, those restrictions generated by more than one node have, in general, a weight equal to one (or equal to the number of nodes that generate this condition). In the next section we see how these alternatives impact in the size of the obtained MPR set.

2.4. Practical implementations

In general, there will always exist several solutions to the MPR selection problem. The minimal MPR set (i.e., the set with minimal cardinality) that solves the selection problem (hereinafter denoted as “min”) can be computed by using a brute force strategy. However, the complexity of this approach is so high that, in practice, it only applies for small networks. The approach relies on simply testing all possible subsets until finding the minimal one that fulfills all the requested conditions. Optimized versions of a brute force selection, based on *search and retry* heuristics, exist [5]. These variants add to the procedure a sorted test of the subsets and rules to prune the search tree. These solutions still present, however, a very costly complexity that renders the process still impractical for average systems.

A more practical solution, based on a greedy selection strategy, is summarized in Function 1. Function 1 gives a solution of a size smaller than $|\min|(1 + \log(|\min|))$ [5], where $|\min|$ refers to the size of the minimal set solution. A worst case example is given in Figure 1. The steps of Function 1, when it is applied to the network depicted in Figure 1, are the following:

- i) In the first iteration, node 2 is selected, since it has eight neighbors in T not yet covered (an amount of neighbors greater than node 0 (=7), node 1 (=7), node 3 (=4), and node 4 (=2)). Restrictions related to nodes 5 to 12 are fulfilled and removed.
- ii) In the second iteration, node 3 is selected (four neighbors).
- iii) In the last iteration, node 4 is selected.

Function 1 computes the set $\text{MPR}(C, T) = \{2, 3, 4\}$. Note that the solution is not minimal. An alternative solution with exactly two nodes ($\text{MPR}(C, T) = \{0, 1\}$) is also valid. This drawback can be handled if the nodes in T that have the same neighbors than a previous one are filtered. In the network given in Figure 1, all white nodes can be filtered by this rule except nodes 5, 9, 13, 15, 17, and 18. If so, Function 1 provides the solution $\text{MPR}(C, T) = \{0, 1\}$ that is, moreover, minimal. It is, therefore, better to apply a greedy strategy counting only the number of different rules that a node fulfills, instead of the total number of rules. In general, if two or more nodes in T generate the same restriction, only one instance of the restriction must be addressed.

An alternative solution is to give priority to those nodes that are poorly covered by the system. For example, in the example depicted by Figure 1, we can observe that some nodes (e.g., node 1) are mandatory, since they have neighbors in T that no other neighbor covers. Thus, a logical first step is the addition to the MPR of this kind of nodes. Then, the following step is to analyze the cases where a node in T has only two possible candidates. It is mandatory that one of these two candidates gets included in the MPR. Therefore, it seems a convenient approach to start solving this restriction before proceeding with *easier* ones (i.e., rules with more possible candidates). If so, a new procedure can be proposed. Function 2 summarizes the new approach. The new selection mechanism tries to fulfill all restrictions starting by the most restrictive one (i.e., smaller number of variables in the rule, i.e., smaller number of nodes in C neighbors of the nodes in T) until the less restrictive rule (e.g., the rule containing the largest amount of variables or candidates). If such a rule has only one associated candidate, the corresponding node is added to the MPR. Otherwise, if the

Function 1. Greedy MPR Selection

1. Start with an empty MPR set.
 2. Look for the candidate c which is present in more restrictions not yet fulfilled by any other candidate. At least one restriction must be found.
 3. The node c is added to the MPR set, remove this node from the candidates list and remove all restrictions already fulfilled.
 4. Repeat Steps 1 and 2 until all restrictions are covered or no more candidates are available.
 5. Return the resulting MPR set.
-

rule has more than one candidate, a heuristic method to select the best candidate applies (e.g., a greedy-like selection strategy).

We can finally close the section by defining the standard MPR selection procedure used in the OLSR protocol as a mixture of *greedy* selection and *most restrictive rule first* selection. An overview of the complete process is summarized in Function 3.

Function 2. Most Restrictive First MPR Selection

1. Start with an empty MPR set.
2. Select the target node in T which has smaller number of neighbors in the candidate set C . Let C' be the set of these neighbors.
3. Look for the node included in C' that has maximum number of neighbors in the target set T .
4. Add the node found in previous step to the MPR set. Remove it from C . Decrease the number of pending MPRs for all their neighbors in T . If the node has reached the requested coverage, remove it from T .
5. Repeat from the first step until T or C are empty.
6. Return the resulting MPR set.

Function 3. Standard OLSR MPR Selection

1. Start with an empty MPR set.
2. Apply Function 2, i.e., *most restrictive rules first*, to select those candidates which appear in a rule of a size that is lower than the m_{cvg} parameter. Add the nodes to the MPR set.
3. Apply Function 1, i.e., *greedy selection*, until all the remainder rules are fulfilled.
4. Return the resulting MPR set.

3. Flooding of broadcast messages

A common strategy for flooding messages on a network is the use of a source-independent procedure (cf. Procedure 1). A subset of nodes is in charge of retransmitting messages. The subset must satisfy that any of its nodes successfully reaches any other node in the network by a path that stays entirely within the subset itself. The cost in resources to flood the network is proportional to the size of the subset. Therefore, the goal is to obtain an optimal subset (in terms of size).

OLSR addresses the problem in a different way. It uses the selection of MPRs that we introduced in Section 2, followed by a source-dependent flooding strategy that works as follows: a node forwards a broadcast message if and only if (1) it is the first time this message has been received by the node and (2) the adjacent sender of the message has selected the node as one of its MPRs. In other words, any message is discarded if it has been previously received or if it is received by non MPR nodes [11]. The method is called *source-dependent*, since a node decides to forward messages by taking into account the origin of such messages. Hence, the total amount of forwards in the network is

Procedure 1. Source-independent Flooding

1. The node which originates the message sends it to all its neighbors.
2. A node which receives the message forwards it to all its neighbors if and only if it is the first time it receives the message AND the node belongs to the predefined subset of forwarders. Otherwise, the message is not forwarded.
3. Step 2 is repeated until no more forwards are executed for the message.

Procedure 2. OLSR's Standard MPR Flooding

1. The node that originates the message sends it to all its neighbors.
2. A node that receives the message forwards it to all its neighbors if and only if it is the first time it receives the message AND the sender of the received message is included in the node list of MPR selectors. If not, the message is not forwarded.
3. Step 2 is repeated until no more message forwards are executed.

considerably reduced. Procedure 2 summarizes the source-dependent flooding process defined in the standard OLSR protocol [4]. Notice that the flooding method proposed in Procedure 2 has an important drawback when applied to faulty networks. Indeed, if delays in the transmission of a message can lead, for instance, to a situation such that not all the network nodes receive the message, the propagation time can produce a failure in the flooding process (even when the retransmission handles the failure of any single message). For instance, let us consider the sample network depicted by Figure 2. If we now as-

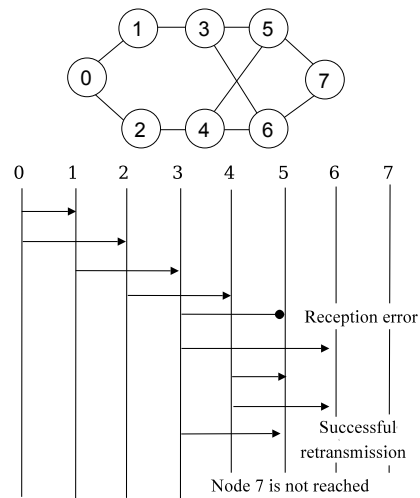


Figure 2: Example of failure in a source dependent flooding.

sume that $\text{MPR}(3) = \{1, 5\}$ and $\text{MPR}(4) = \{2, 6\}$, we observe that any error or delay in the reception of a given message may cause that some network nodes do not receive the message they intend to flood (even when such a message has been successfully retransmitted).

The aforementioned problem of delays does not affect a source-independent flooding. Therefore, it is possible to improve the flooding mechanism of OLSR by combining source-independent flooding and multipoint relaying. The approach presented by Adjih *et al.* in [1] is a proper example of this combination. The solution proposes the transformation of the original MPR structure into a *Connected Dominating Set* (CDS). A CDS derived from a set V of nodes is defined in the following equation

$$N_1^*(\text{CDS}) = V \wedge \text{connected}(\text{CDS}),$$

where a recursive definition of connected is:

$$\begin{aligned} &\text{connected}(\{x\}), \\ &\text{connected}(X \cup \{y\}) \leftarrow \text{connected}(X) \wedge y \in N_1(X). \end{aligned}$$

By satisfying the properties of a valid CDS, the transformation allows the use of a source-independent flooding strategy while keeping the total amount of forwards in the network considerably low. The transformation method in [1] requires a numeric labeling of the nodes. This can be based, for instance, on the IP address of the nodes. Let N_{\min} be the smallest node identifier of a neighborhood. Then, the Adjih *et al.*'s method selects as forwarders of the underlying CDS those nodes that satisfy the following condition:

$$\begin{aligned} &\text{The node identifier corresponds to } N_{\min} \\ &\quad \vee \\ &\text{The node is in the MPR of an adjacent node with } N_{\min}. \end{aligned}$$

A variation of the previous method was proposed by Wu *et al.* in [14, 15]. The first part of the previous *or* condition is slightly modified to reduce the CDS size:

$$\begin{aligned} &\text{(The node identifier corresponds to } N_{\min} \\ &\quad \wedge \\ &\text{The node has, at least, two unconnected neighbors)} \\ &\quad \vee \\ &\text{The node is in the MPR of an adjacent node with } N_{\min}. \end{aligned}$$

The construction of this CDS-based forwarding subset, as a replacement of the multipoint relying set of any standard OLSR system, is claimed by the authors as a promising trade-off between minimization of forwarders and reliability of the exchanged information. Moreover, an increased flood redundancy can be achieved by these methods if an MPR coverage factor greater than one is settled during the MPR selection state. In Section 5, we show that this increase leads, however, to a considerable performance overhead. In the sequel, we provide a new CDS-based flooding method, also compatible with the addition of a dynamic MPR coverage parameter, that improves the performance of the standard OLSR flooding procedure, as well as the performance of the CDS-based flooding methods proposed in [1, 14, 15].

4. Flooding method proposal

In Procedure 3, we propose a new flooding method that optimizes the flooding processes surveyed in Section 3 and that allows the addition of extra coverage (as suggested by Cuppens *et al.* in [7] for robustness reasons). Our solution gets inspiration from the use of *mathematical*

morphology methods applied to graph optimization [13]. In this sense, our method generalizes the notion of *multipoint relaying*, by adding distance knowledge. This way, it improves the selection of an optimized subset of forwarders, organized as an underlying CDS flooding structure. It implements, moreover, source independent flooding of broadcast messages, handling the problem of delayed forwards pointed out in Section 3 (and which is an inherent limitation of the source dependent flooding method proposed in the OLSR protocol).

Procedure 3. Distance-enabled CDS Flooding

1. Select one node of the network s .
 2. Obtain the maximum $\text{dist}(s, j)$ for all nodes, d_{\max} . Notice that the case $d_{\max} = 0$ (i.e., a single node network) can be ignored. Let a variable d with initial value $d_{\max} - 1$.
 3. Define a node set P of pending nodes and initialize it with all network nodes.
 4. Find the solution of $\text{MPR}_d(s) = \text{MPR}(N_d(s), N_{d+1}(s))$ using any of the methods described in Part I of this document.
 5. All network nodes j at a distance d from s and which are dominated by previous set $\text{MPR}_d(s)$ but do not belong to it are removed from P :
 $j \in N_1(\text{MPR}_d(s)) \wedge j \notin \text{MPR}_d(s) \wedge \text{dist}(s, j) = d$.
 6. If $d > 0$ then repeat steps 4 and 5 for $d \leftarrow d - 1$.
 7. The resulting CDS is the union of all previously computed MPR subsets, that is :
 $\text{CDS} = \text{MPR}_0(s) \cup \dots \cup \text{MPR}_{d_{\max}-1}(s)$.
-

4.1. Sample construction

Assume the sample network depicted in Figure 3. In the sequel, we summarize the execution steps of Procedure 3 in order to compute the underlying CDS-flooding structure for such a network.

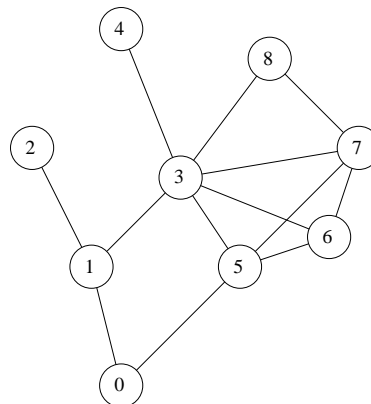


Figure 3: Sample network with node identifiers.

Figure 4 shows the distance (in hops) from a node selected as origin (node 0 in this example) to all other nodes. The computation of these distances requires the knowledge of all the links in the system

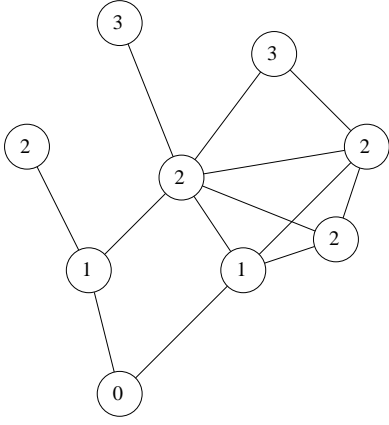


Figure 4: Procedure 3, Step 2. Distances to node 0.

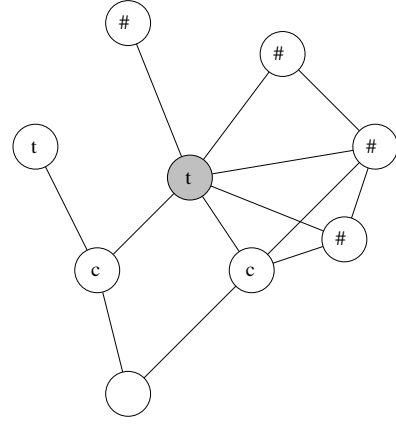


Figure 6: Second sequence, $d = 1$.

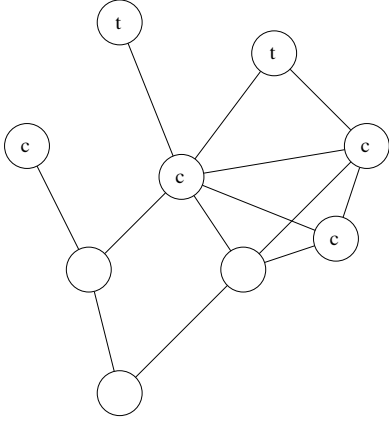


Figure 5: First sequence, $d = 2$.

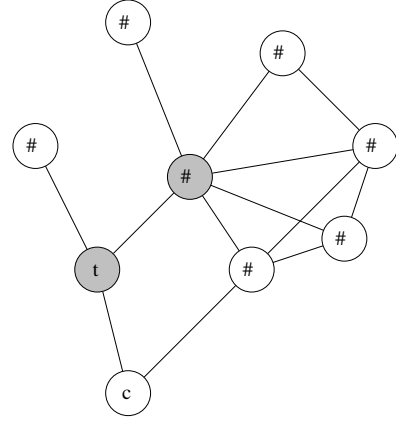


Figure 7: Final (trivial) sequence, $d = 0$.

(already provided by an original construction of the OLSR system using multipoint relaying flooding). The maximal distance computed is $d_{\max} = 3$. At distance $d = 3$, there are two nodes that are selected as the targets of an MPR problem. The nodes at distance $d = 2$ are the candidate nodes for this problem. In Figure 5, target nodes are labeled t and candidate nodes c . Any of the MPR selection functions defined in Section 2 can now be used to solve the problem with $T = \{4, 8\}$, $C = \{2, 3, 6, 7\}$. The result is an MPR set equal to $\{3\}$. This is the first subset of nodes to be included in the final CDS.

The next sequence handles nodes at distance $d = 2$ (cf. Figure 6). At this distance, we have four nodes. Two of them are already neighbors of a node in the CDS (nodes labeled by the symbol $\#$ in the figure). Thus, the target nodes of the next MPR problem are the nodes in $T = \{2, 3\}$. The candidate nodes are now all the nodes at distance $d = 1$, *i.e.*, nodes in $C = \{1, 5\}$. Again, any valid MPR selection function is used to solve the problem that results in the set $MPR = \{1\}$. This subset is added to the final CDS. Then, the procedure continues by handling the nodes at distance $d = 1$ (cf. Figure 7). At this distance, the solution of the problem is now trivial. Note that although there are two nodes, only one of them is already neighbor of a node in the CDS. Thus, the only target node of the next MPR problem is in $T = \{1\}$. The only candidate node at distance $d = 0$ is in $C = \{0\}$. The result is $MPR = \{0\}$. Therefore, node 0 is added to the CDS. Figure 8 shows the final result, *i.e.*, $CDS = \{0, 1, 3\}$.

We recall that, like in [1, 14, 15], our method requires an initial stage in which a standard OLSR *multipoint relaying flooding* initially

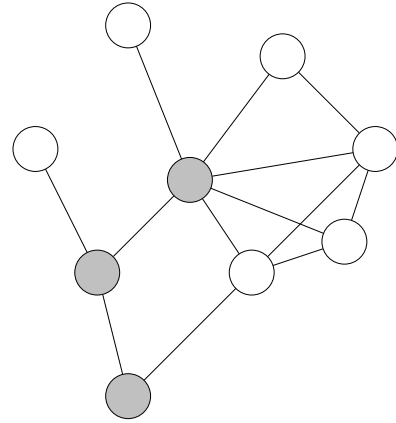
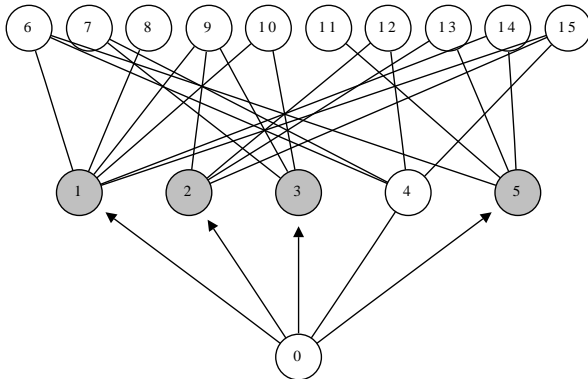


Figure 8: Resulting CDS subtopology derived from Procedure 3.

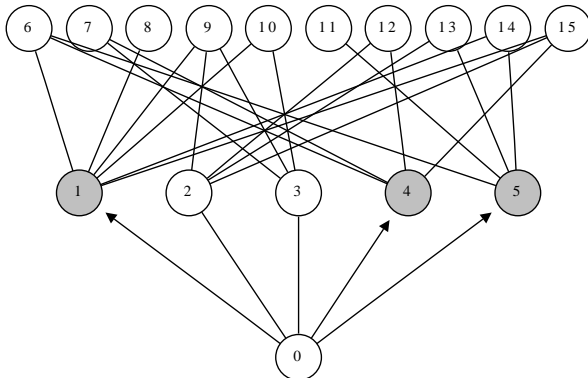
sends TC messages to provide the list of links to each node in the system. The completeness of this knowledge may impact in the quality of the obtained CDS. Let us also observe that a reduction in the number of nodes of the CDS has a clear impact in the efficiency of the flooding process (specially when nodes in the network are sparse). Thus, in sparse distributions of nodes, and when extra coverage is required, some nodes are likely going to be poorly covered. If that occurs, it can easily be solved by increasing the MPR coverage parameter.

5. Experimental simulations

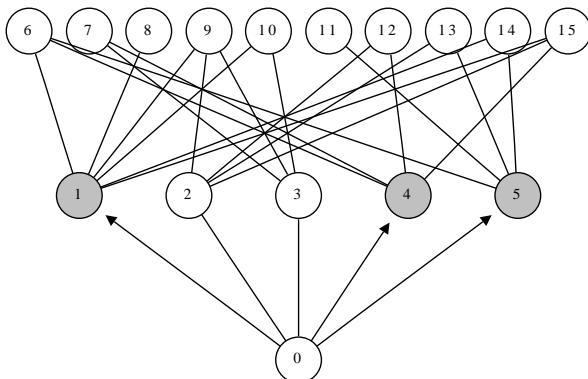
This section presents experimental measurements that are performed to analyze and confirm the main characteristics of the processes described in previous sections. The experiments are executed in a simulation environment. Different constructions, with different densities of nodes and range of communication, are deployed in a simulated unit square area. The deployment of nodes follows a random pattern with a uniform distribution in the x and y axes. The different MPR selection functions and flooding procedures are then applied to these networks. The series of experimental simulations are implemented using the Objective Caml [6] programming language.



a) Greedy Selection (cf. Function 1)



b) Most Restrictive First Selection (cf. Function 2)



c) Standard OLSR selection (cf. Function 3)

Figure 9: Example of MPR set constructions based on three different MPR selection methods. Gray nodes represent the nodes selected as MPR nodes.

5.1. Evaluation of the MPR selection functions

The objective of our two initial simulations is simply to measure the average number of nodes included on the MPR sets computed by the functions we defined in Section 2. A sample testbed (with five candidates and ten target nodes) is depicted in Figure 9. The node that calculates the MPR sets in all three examples is node 0. The gray nodes represent the nodes selected by node 0 as MPRs. Each set of experimental simulations contains thirty candidate nodes and one hundred target nodes. All target nodes are, at least, linked to one candidate node. Additional links between targets and candidates nodes are created at random. A network definition parameter, l , is used to fix the average number of links. It takes values from 0.4 (meaning more links between candidates and targets) to 0.15 (meaning a small number of links). One hundred different networks have been tested and averaged for each value. Table 1 summarizes the results of the experimental simulations. Notice that the three MPR selection methods provide almost the same cardinality.

MPR selection method	Average size of MPRs
Function 1 (Greedy)	12.95
Function 2 (Most restrictive)	12.73
Function 3 (OLSR)	12.83

Table 1: Average MPR size vs. selection functions.

In a second experimental simulation, we measure the MPR coverage parameter (m_{cvg} for short), applied to the standard OLSR selection of MPRs (cf. Function 3), and its effects on the size of the resulting MPRs. We do not show the results for Functions 1 and 2, given that the differences are minimal. The experimental setting is the same as in the previous series of simulations. The MPR coverage factors are set to 1, 2, 3, and 4. Ten different networks are tested for each set of parameters. The results of these tests are plotted in Figure 10. From the results of this second series of simulations, we conclude that: (1) an increase of the MPR coverage factor in one unit increases by, almost, a 50% the size of the resulting MPRs; and (2) that not all the nodes handle the requested MPR coverage, *i.e.*, there are always a considerably high amount of nodes that are poorly covered.

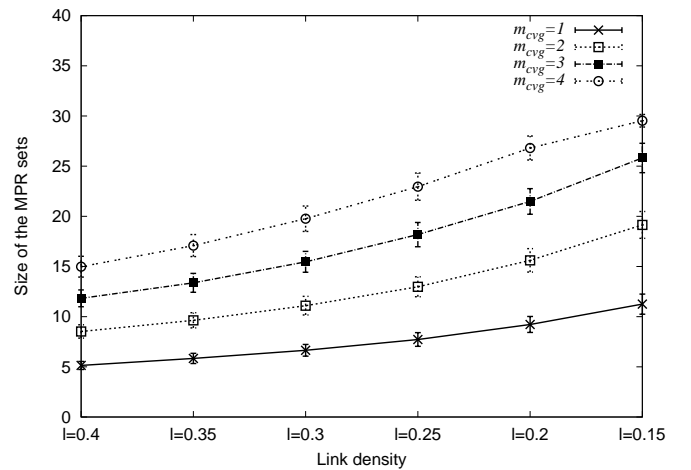


Figure 10: MPR sizes for the OLSR selection (Function 3) with extra coverage.

5.2. Handling delays with the standard MPR flooding

The objective of the following simulation is to analyze the sensibility of the standard MPR source dependent flooding method (cf. Procedure 2) of the OLSR protocol, when messages require extra retransmission due to reception errors. Indeed, the simulations measure the ability of the OLSR procedure in achieving this task. Each simulation creates several random networks and computes the retransmission of TC messages on each network. The retransmission of the messages from a node towards its neighbors is conditioned by a probability of failure (provided as input parameters for each simulation). In real scenarios, we can assume that this failure might be caused by situations such as interferences, data corruption, or packet collision in wireless networks. In case of failure, the original message is discarded and its retransmission is queued. Although the message is never lost, this retransmission produces a delay in the reception from some network nodes. This is the way how the simulations analyze the sensibility of the flooding procedure to the message delays.

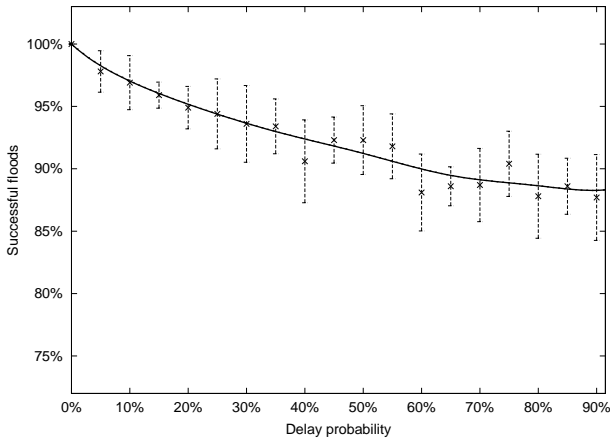


Figure 11: Standard OLSR flooding with delayed TC message retransmissions.

Figure 11 depicts the results of ten thousand simulations, with different network definition parameters and delay probabilities. The flooding process is considered to fail when, at least, one network node does not receive the message. As expected, an increase in the probability of message delay produces an increase in the number of networks for which the flooding mechanism fails. We, therefore, conclude that if the physical medium of the network can produce this kind of transmission artifacts, the use of the source dependent flooding strategy used by the OLSR protocol must contain message acknowledgments or other protection techniques in order to assure that all the information is properly flooded.

5.3. Forwarding evaluation for each flooding method

The following series of simulations compare the number of forward nodes when using the different flooding procedures. In all the simulations, n nodes are randomly placed in a unit square area. If a node becomes isolated from all the remaining nodes and disconnected from the source of the flooding, it is ignored. All the nodes have the same communication range (denoted here as r). Bidirectional links between each pair of nodes closer than r are available. The source of the flooding is always the node with identifier 0. The simulations evaluate the flooding method of the standard OLSR protocol (cf. Procedure 2), the

MPR-based CDS flooding methods proposed in [1, 14, 15] (cf. Section 3), and our proposal (cf. Procedure 3). The results of the simulations are shown in Figure 12. The average number of nodes retransmitting the message is 41.33 nodes with the MPR flooding method, 41.89 for MPR-CDS method proposed by Adjih *et al.* [1], 38.37 for MPR-CDS method proposed by Wu *et al.* [14, 15], and 20.54 for our proposed flooding method. Notice that our proposal presents a saving of resources of almost a 50%. It also reaches a stable number of forwarders when the density of the network is high enough to cover most of the unit square area. See, for example, the differences from $r = 0.2$, $n = 50$ to $n = 100$ and to $n = 150$, where the number of forwarders when using our proposal remains stable whereas, in the other methods, this number continues increasing proportionally to n . Similarly, our approach also reaches a stable number of forwards for combinations $r = 0.15$, $n = 150$ and $r = 0.15$, $n = 300$.

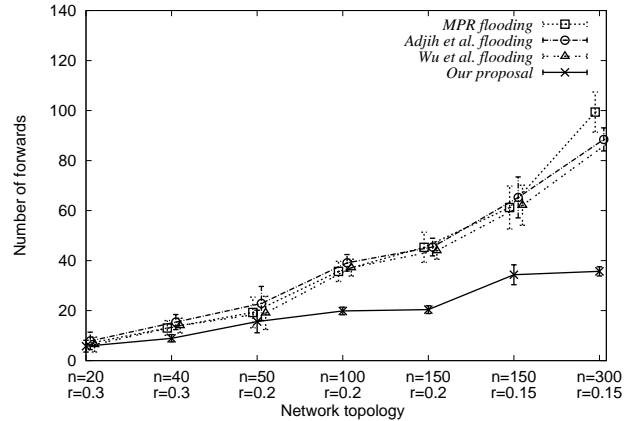


Figure 12: Forwarding evaluation for each flooding method (with $m_{cvg} = 1$).

The following series of simulations continue the evaluation of the flooding methods when the MPR coverage parameter increases up to a factor of four. We recall that none of the methods can guarantee that the required MPR coverage factor settled with values greater than one

Method	MPR coverage	Cost ratio
MPR flooding [4]	1	35%
Adjih <i>et al.</i> [1]	1	41%
Wu <i>et al.</i> [14, 15]	1	37%
Our proposal	1	21%
MPR flooding [4]	2	56%
Adjih <i>et al.</i> [1]	2	58%
Wu <i>et al.</i> [14, 15]	2	54%
Our proposal	2	35%
MPR flooding [4]	3	68%
Adjih <i>et al.</i> [1]	3	69%
Wu <i>et al.</i> [14, 15]	3	65%
Our proposal	3	45%
MPR flooding [4]	4	73%
Adjih <i>et al.</i> [1]	4	75%
Wu <i>et al.</i> [14, 15]	4	71%
Our proposal	4	52%

Table 2: Forwarding evaluation for each flooding method.

will be handled by all the nodes. Table 2 summarizes the complete series results. The cost criteria that we take into account is the amount of forwarded messages that are generated for every method, normalized by the network size (denoted in Table 2 as *Cost ratio*). Observe that in all four cases, our proposed method turns on a number of retransmissions that is much smaller than all the other flooding methods.

5.4. Distance-enable MPR flooding with partial knowledge

Most of the surveyed flooding procedures use only local information about the network, *i.e.*, the nodes need to know only their neighbors at one or two-hop distance. Some other methods need a global network overview, like all link-state protocols. The input of these processes is the link information database. This database can be complete or partial. Incomplete link database means a partial knowledge of $N_d(x)$ which reduces the quality of the method. The amount of information which a node stores about network links depends on the network protocol. The usual situations are as follows:

- All nodes know all links.
- In protocols like OLSR, only the MPRs do forward the topology control (TC) messages. This means that only links from/to a node and their selected MPRs are known by any node.
- MPR nodes broadcast link information about all their neighbors, not only about the ones which selected them as MPRs. This option does not increase the number of control messages, only their size.

When using our proposed method, it is relevant to measure how the different amount of TC information available at each node affects the performance of the flooding procedure. Figure 13 shows how the differences in the amount of available information affect the number of forwarders. Several networks with different sizes (denoted as n) and communication ranges (denoted as r) are evaluated.

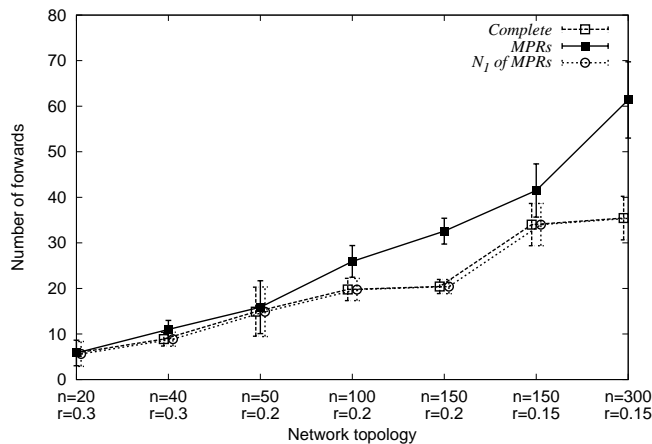


Figure 13: Number of forwards when link database is not complete.

Note that knowledge of full network information or knowledge of only MPR neighbors (much better in the amount of control messages) has a very similar performance in the flooding. In addition, a minimal OLSR compliant network where nodes know only the links from MPR selectors to MPRs have a penalty of about 50% more forwards and the number of forwarders is, almost, the average between the optimal case and the standard OLSR method.

6. Conclusion

We have analyzed the formal description of the MPR subset problem defined in the OLSR protocol. We have presented some relevant methods for solving this problem, as well as some existing methods based on the MPR concept to perform an optimized flooding of messages over OLSR-based systems. We have then proposed a new flooding mechanism for such systems. Simulations show that our proposal improves the performance of previous existing solutions. Indeed, it provides remarkable results, since it significantly decreases the use of network resources when flooding broadcast messages over the system. Full adaptation of our proposal in some similar protocols, taking into account their intrinsic differences and limitations, is a next step in our research to confirm the benefits of the approach that we have reported.

Acknowledgments

This work is partially supported by the Spanish Ministry of Science and Innovation and the FEDER funds under the grants CONSOLIDER-INGENIO 2010 CSD2007-00004 ARES and TSI2007-65406-C03-03 E-AEGIS, and the Institut TELECOM (programme Futurs et ruptures).

References

- [1] C. Adjih, P. Jacquet, and L. Viennot. Computing Connected Dominated Sets with Multipoint Relays. *Ad Hoc & Sensor Wireless Networks*, March 2005.
- [2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless Sensor Networks: A Survey *IEEE Computers Networks*, 38(4):393–422, 2002.
- [3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks *IEEE Communications Magazine*, 40(8):102–114, 2002.
- [4] T. Clausen and P. Jacquet. Optimized Link State Routing Protocol (OLSR) IETF RFC 3626, October 2003.
- [5] G. Călinescu, I. I. Măndoiu, P. J. Wan, A. Z. Zelikovsky. Selecting forwarding neighbors in wireless ad hoc networks Misbehaviors detection to ensure availability on OLSR. *Mobile Networks and Applications*, 9(2):101–111, 2004.
- [6] Institut National de Recherche en Informatique et en Automatique (INRIA). The Caml Language. <http://caml.inria.fr>
- [7] F. Cuppens, N. Cuppens-Bouahia, T. Ramard and J. Thomas. Misbehaviors detection to ensure availability on OLSR LNCS 4864, pages 799–813, 2007.
- [8] J. Harri, C. Bonnet, and F. Filali. OLSR and MPR: Mutual Dependences and Performances. Research report RR-05-138, Institut Eurécom, March 2005.
- [9] U. Herberg and T. Clausen. Security Issues in the Optimized Link State Routing Protocol Version 2 (OLSRv2). *International Journal of Network Security & Its Applications (IJNSA)*, 2(2):162–181, April 2010.
- [10] U. Herberg. Performance Evaluation of using a Dynamic Shortest Path Algorithm in OLSRv2. *8th Annual Communication Networks and Services Research (CNSR)*, pp.376-380, IEEE Communications Society, Canada, May 2010.
- [11] A. Qayyum, L. Viennot, and A. Laouiti. Multipoint Relays for Flooding Broadcast Messages in Mobile Wireless Networks. In: 35th annual Hawaii international conference on system sciences, 2002.
- [12] K. Sohrabi, J. Gao, V. Ailawadhi, and G. Pottie. Protocols for self-organization of a wireless sensor network *IEEE Personal Communications*, 7(5):16–27, Oct 2000.
- [13] L. Vincent. Graphs and mathematical morphology. *Signal Processing*, 16(4):365–388, Elsevier, 1989.
- [14] J. Wu. An Enhanced Approach to Determine a Small Forward Node Set on Multipoint Relays. *IEEE Vehicular Technology Conference*, 2003.
- [15] J. Wu, W. Lou, and F. Dai. Extended Multipoint Relays to Determine Connected Dominating Sets in MANETs *IEEE Trans. On Computers*, vol.55, 2006.