Institut Eurécom
Department of Mobile Communications
2229, route des Crêtes
B.P. 193
06904 Sophia-Antipolis
FRANCE

Research Report RR-05-138

OLSR and MPR: Mutual Dependences and Performances

March 23^{rd}, 2005

Jérôme Härri, Christian Bonnet and Fethi Filali

Tel : (+33) 4 93 00 26 26
Fax : (+33) 4 93 00 26 27
Email : {Jerome.Haerri,Christian.Bonnet,Fethi.Filali}@eurecom.fr

---

[1] Institut Eurécom's research is partially supported by its industrial members: Bouygues Télécom, Fondation d'entreprise Groupe Cegetel, Fondation Hasler, France Télécom, Hitachi, Sharp, ST Microelectronics, Swisscom, Texas Instruments, Thales.

OLSR and MPR: Mutual Dependences and Performances

Jérôme Härri, Christian Bonnet and Fethi Filali

Abstract

Optimized Link State Routing protocol (OLSR) is one promising routing protocol for mobile ad hoc networks. It recently reached the RFC step and is still under improvement before becoming a standard. Since its first design, OLSR has been associated with the Multipoint Relay (MPR) protocol to reduce the flooding of OLSR topological messages. Many papers has been written on possible promising solutions to improve OLSR by replacing MPR by another topology control protocol, modifying MPR heuristic, or adapting OLSR to fast mobility and include security. But few papers have dealt with the interactions between MPR and OLSR and what makes them so particular. In this paper, we analyze OLSR requirements and MPR properties and show that OLSR optimality is bound to the deep cooperation and interaction between MPR and OLSR. We specifically exhibit the intrinsic properties of MPR absolutely necessary to OLSR, and that cannot be found in any other topology control protocol at this time. Conversely, we illustrate how OLSR is able to correct MPR's drawbacks to reach its objective. We also show that OLSR suffers from convergence problems that are commonly dealt with adequate configuration parameters as suggested in the RFC3626. Finally, we suggest that any attempt to alter any of the two protocols, or their respective configuration parameters, will lead to suboptimal results.

Index Terms

Optimality, convergence, MPR, OLSR, ad hoc network.

Contents

List of Figures

1 Introduction

Mobile Ad Hoc Networks (MANETs) is an emergent concept in view for infrastructure-less communication. Actual technologies are based on a backbone of access points, which mobile devices can connect to.Yet, a growing demand on increased bandwidth and improved communication quality made engineers choose to decrease the transmission range of mobile remote devices, since higher radio link capacity implies shorter radio ranges in ground communication. Consequently, the backbone needed to be re-designed with an increased number of access-points, which starts to penalize this concept in term of deployment capabilities.

MANETs, on the other hand, do not require such infrastructures and hence promise to be more easily deployable than infrastructure-based networks. Since communications do not rely on any coordinator, collisions and packet losses increase as the density of the network. Exchange of messages should be therefore limited to the strict necessity. Moreover, as the transmission range decreases and the number of hosts increases, routing protocols need to be robust in order to insure a fully connected network.

Initially, routing protocols has been adapted from wired networks. However, in these networks, flooding (distributing information to each and every node in the network in an uncontrolled way) was a key part of information dissemination. When the network is dense, the overhead due to this kind of information dissemination may become prohibitive. Despite its simplicity, flooding is very inefficient and can result in high redundancy, contention and collision. This is the main motivation for many research teams that have proposed more efficient flooding techniques whose goal is to minimize the number of retransmissions while attempting to ensure that a broadcast packet is delivered to each node in the network. Difference approaches of flooding techniques and broadcasting control protocols exist and are listed in [1]. To our knowledge, Reactive Routing Protocols, or On-Demand, only use limited broadcasting control features. However, Proactive, or table-driven, routing protocols rely deeply on flooding reduction to limit their control overhead and reach scalability. Therefore, Proactive protocols and Flooding reduction algorithms need to collaborate, each one doing its particular task, with the objective to create optimal routes at a low control overhead.

Multipoint relays (MPR) [2] provide a localized and optimized way of flooding reduction in a mobile ad hoc network. Using 2-hop neighborhood information, each node determines a small set of forward neighbors to relay messages. By doing so, MPR avoids multiple retransmissions and limits the number of nodes relaying traffic to only a few. Basically, MPR is part of a proactive routing protocol called Optimized Link State Routing (OLSR) [3]. MPR's role is to reduce flooding of Topological Control (TC) messages needed by OLSR to create optimal routes in the network. Depicted like this, one might think that both protocols are completely separated and could even be independently tested, improved, or even changed. This is the case for many protocols. However, OLSR has a much different relationship with MPR.

In this paper, we will explain that, since OLSR and MPR have been commonly developed, they deeply interact among each other. We will show that MPR has been specifically designed for OLSR and OLSR for MPR, and that replacing any one of them leads to performance reduction. In other words, for **OLSR** and **MPR**, **complementarity leads to optimality**. We will also point out some convergence issue of OLSR and MPR, and argue that they are compensated by proper configuration parameters which alteration will lead to decrease in performances. Finally, we will exhibit a possible solution, to the best of our knowledge not yet proposed, to improve MPR's convergence by reducing its ties to critical message exchanges during the process of MPR election.

The rest of the paper is organized as follows. Section 2 lists OLSR requirements and particularities, while Section 3 describes MPR properties and the heuristics proposed for MPR nodes election. In Section 4, we show that several attempts to improve OLSR using meshed-based Connected Dominating Set (CDS) flooding does not improve OLSR's performance, and in Section 5, we give similar considerations for tree-based CDSs. Section 6 exhibits some convergence issues in OLSR and MPR, while Section 7 draws some concluding remarks and proposes a novel approach to improve OLSR.

2 OLSR

In order to compute and maintain routes from and to any nodes in a mobile ad hoc network, OLSR performs for each node a loop discovery on every path to any nodes in the network. Therefore, at convergence, each node keeps a routing table that indicates the next hop node to reach any destination node. This path is unique[1] and is loop-free. So as to perform this task, every node periodically broadcasts topological messages containing a list of Multipoint Relay Selector, or a list of neighbors that chose it as MPR. Since these TC messages are broadcast to the entire network, a serious flooding control mechanism needs to be implemented. Consequently, only a subset of nodes is used to reduce the broadcast storm problem. This technique is usually called topology management and finds its origin to the graph theory. Based on a fully connected graph, one need to find a subset of nodes, called *Dominator*, that forwards packets while others simply listen. The set of all dominators is referred as a *Dominating Set (DS)*. Several algorithms in graph theory are aimed to reduce the DS to reach a *Minimum Dominating Set (MDS)*. Unfortunately, such problem is *NP-hard* [4, 5]. Taking MPR into consideration, every multipoint relays may be considered as a Dominator and the set of all MPRs represents a DS. Yet, this DS is far from being optimal, and it is usually OLSR's role to reduce this set to a subset of relays and tries to create a *Minimal Connected Dominating Set (MCDS)*, where any node is either a dominator or is adjacent to a dominator. Similarly, this is a well-known *NP-Complete* task, even if the complete topology is available.

---

[1] we do not consider alternate route for robustness

After these considerations, let us study the OLSR's requirements. In order to reduce the broadcast storm due to the TC messages, OLSR needs a DS, not specifically minimum. This is a very important point that we want to emphasize here. The most important property OLSR needs from a topology management protocol for broadcast reduction is its low fraction of nodes implied in a DS flooding. In other words, no matter how big is the DS, only the number of dominator effectively requested for flooding is of a particular significance. Secondly, since OLSR will try to reduce the DS, it is necessary that any removed dominator does not trigger a complete topology management update, as it could be the case in a tree-like topology. Finally, as mentioned above, creating a MCDS is *NP-complete*. Therefore, reducing the number of dominator while keeping a full connectivity might reveal to be an impossible task. Thus, another approach would require a suboptimal CDS in which dominators do not blindly relay packets but perform a selective relaying. Thanks to this trick, a sub-optimal CDS would tend to act as an optimal one.

3 MPR

In the previous section, we illustrated the OLSR's requirements to be fully functional. Let us now analyze the MPR protocols and its particular properties. Initially, MPR used the following greedy algorithm.
Let us consider MPR at node i

- Begin with an empty MPR set
- First Step: Select those one-hop neighbor nodes of i that are the only neighbor of some two hop neighbors of i, and add them to the MPR set.
- Second Step: Add in the MPR set the neighbor node of i that covers the largest number of two-hop neighbors of i that are not yet covered by the current MPR set. Repeat this step until all two-hop neighbors are covered.

Authors in [10] found that 75% of all MPRs were elected during the first step. Although being optimal, this step makes the total number of iteration of $O(m^2)$. Therefore, in was interested to try to find another heuristic that could work in $O(m)$ steps.
This approach was called *min-id*.

- Begin with an empty MPR set
- Check the neighbor nodes in the increasing order of their identifiers, If the current node covers a two-hop neighbor that was not yet covered by the current MPR set, then add it the MPR set.

Although the min-id algorithm works in $O(m)$ steps, it is far from optimal. Indeed, authors in [6] showed that the average number of MPR per nodes in the min-id algorithm was far larger than the one obtain by the original heuristic, since it eliminate non optimal nodes that would be inserted in the min-id algorithm. Thus, the MPR selector set per node is bigger and any node will have more chance to have to relay a node. This is exactly what we want to avoid when trying to improve OLSR. So, the original heuristic is more optimal and more specifically, one should not neglect the first step of this algorithm.

4 Mesh-based Connected Dominating Set for OLSR

In order to improve OLSR, we are still looking for improvement for the topology management part. We showed in the previous section that, as far as MPR was concerned, the original heuristic was optimal. Then, if we want to improve performances, we might think of changing the MPR protocol for a *better* one. MPR has been designed for flooding reduction, and as we will see later, is very good at this particular task, but does not optimize the number of MPR nodes. Yet, when designing a topology management protocol,we also want to obtain a minimal number of relays. Moreover, MPR requires the last hop node ID in order to decide whether or not a packet should be retransmitted, which could be problematic when unique IDs are not available. Therefore we could think of CDS flooding instead of MPR flooding. Consequently, we should try to use graph theory to design a protocol capable of creating a MCDS. However, as mentioned before, obtaining a MCDS is *NP-Complete*.

A sub-optimal solution to the MCDS, called MininalCDS, has the same properties that the MCDS, yet with a larger set of dominators and links. There are a large number of teams that proposed heuristics that solve this task: Das et al. [11, 12, 13], Wu at al. [15, 19], Stojmenovic et al. [14], to name only a few. In order to keep the MPR assets while improving its drawbacks, Jacquet et al. [6] tried to use MPR as a starting point in order to obtain a MinialCDS. Basically, as they explained in their paper, the first MinimalCDS they could imagine was the set of all MPR nodes. Unfortunately, their results on this particular CDS showed that it contained too many nodes, and was therefore suboptimal. They then proposed MPR relay reduction consisting of rules and that needs the availability of a total order of nodes.
After completion of the MPR algorithm, a node decides that it is in the CDS if

- the node has a smaller ID than all its neighbors, or
- it is MPR of its neighbor with the smallest ID.

The authors then compared their protocol with MPR flooding and the results they obtained were somehow surprising. As expected, the number of dominators was much smaller with the CDS than with MPR. But the most astonishing result came from the average fraction of nodes implied in an MPR flooding compared to the average fraction of nodes belonging to the CDS. It is almost identical. This shows that a CDS is of no use to improve flooding. MPR is optimal. But, how could it be since the number of relays are so much larger that for the CDS ?

The reason is in fact quite simple and once again shows the particularity of MPR. Flooding in CDS and MPRs are done in two difference ways:

CDS forwarding rule: A node retransmits a packet only once after having received the packet and if it belongs to the CDS.

MPR forwarding rule: A node retransmits a packet only once after having received the packet the first time from a MPR selector.

Consequently, MPR relays packet only from a subset of its neighbors, while the CDS rule makes nodes relay packets from all of their neighbors. This is what makes flooding so optimal in MPR. An interesting conclusion may be obtain here:

The number of dominators (or relays) is not the most important property of topology management protocols for solving the broadcast storm problem. How nodes decide upon retransmission is far more important.

To conclude this section, as mentioned in Section 2, a key property for OLSR for being optimal is to have a good flooding reduction protocol and at a low overhead. Since MPR overhead is small (O(n)) and is optimal in term of flooding reduction, a CDS would be of no use in order to improve OLSR, as it comes at the cost of increased overheads (the CDSs proposed in [6] or in [20] are both of $O(m^2)$, where m is the average nodal degree) with no improvement in term of flooding reduction. This is more or lest the same kind of conclusions jacquet [20] came to when he tried to implement another CDS protocol called NCDS to compare CDS performances in OLSR.

Another conclusion that Jacquet wrote in [20] is that OLSR's advertised link state information needs to be kept to a minimal level. OLSR does not only build its optimal routes based on advertised relaying nodes, but on advertised candidates for relaying. In other words, OLSR does not need to have a list of *MPR* nodes, but a list of *MPR Selector* node, or nodes that request other particular nodes to relay their traffic. Therefore, since CDS-based topology management is not configured to this task, such approach will not be appropriate for OLSR. The only way to keep route optimality, and avoid cycles, is to advertise MPR selector links and not dominators. This remark is the final conviction that a CDS algorithm replacing MPR, or on top of MPR, would not be useful, even worse, it could lead to performance reductions for OLSR.

5 Tree-based Connected Dominating Set for OLSR

In this section we draw some remarks on the ineffectiveness of tree-based CDSs' implementations such as MST [7], LMST [8], MST-LMST [9] to improve OLSR. Besides, our remarks may also be relevant to optimal mesh-based CDS.

As mentioned in Section 2, another key requirement for OLSR is to be able to remove unnecessary dominators without triggering a topological reorganization. This is particularly the case in tree-based networks since removing an internal node triggers a breach in connectivity and splits the tree in two sub-trees that will have to somehow reconnect themselves. This reorganization comes naturally at a cost and this should particularly be avoided since OLSR removes a large amount of dominators. Therefore, in order for OLSR to work well, any topology management protocol, or flooding reduction algorithm, has to contain redundancy. Since tree-based CDS are optimized to only contain the limited number of dominators for full connectivity, such approach is clearly inappropriate for OLSR. MPR, in other hand, contains too many relays that have always been seen as a drawback for topology management, and that have been a topic for intensive research for optimization. The point here is, if OLSR works so well, it is that it counts on MPR redundancy to be able to optimize its routes. Therefore, tree-based CDSs (and optimal mesh-based CDS in a certain sense) are more of a drawback than an asset to OLSR.

Nevertheless, tree-based CDSs are optimal for flooding since they contain a similar property as MPR: *selective relaying*. Internal nodes relay traffic based on a particular direction. If it comes from leaves nodes, it will relay to neighbors in the direction of the root node and vice and versa. Unfortunately, this particular feature is not sufficient to compensate its drawback for OLSR dominator reduction. To our opinion, a tree-based CDS would not be appropriate for OLSR for broadcast reduction.

A final consideration here is gateway flooding. Baccelli and Jacquet [16] compared MPR flooding with Gateway flooding. Note that Jacquet [6] already compared MPR flooding with CDS flooding and concluded that CDS flooding and MPR flooding were quite similar. Gateway flooding is a broadcast technique that is extracted from the ad-hoc routing protocol DDR [17]. The protocol uses a forest of logical trees interconnected between them via a set of gateway nodes. They showed through mathematical analysis and simulations that MPR flooding presents a much better optimization than the Gateway flooding.

6 Convergence Issues in MPR

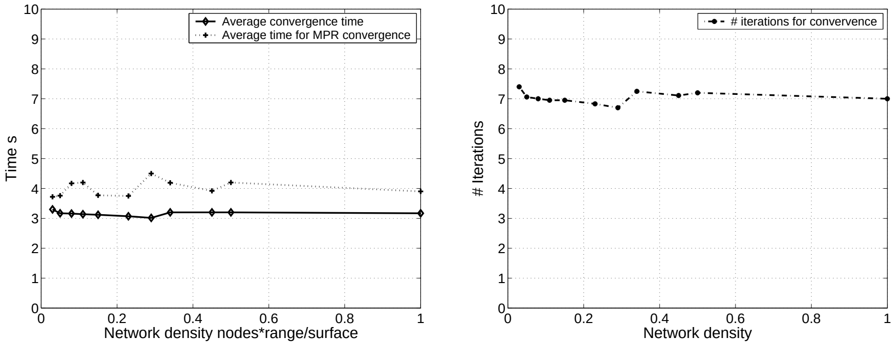
Until now, we discussed on the optimality of OLSR and that any attempt at improving it would be unsuccessful. In this section, we discuss one particular issue in OLSR that has long been occulted: *its convergence*. Authors in [10] claimed that 75% of MPRs were elected on the first MPR step and during the first logical iteration. However, as we will see, this usually requires several physical steps for nodes to obtain correct neighborhood information and accurately elect MPR. In between, we find suboptimal MPRs, links and neighborhood inconsistency. A more alarming convergence issue, which may occurs when the election has been correctly performed, is the Very Important Packets (VIPs) losses. They represent packets containing logical status of links, in other words, nodes that has been elected as MPR, or have lost their MPR status. VIPs losses due to synchronous transmissions, buffer overflows or other channel considerations lead, as we will see later, to network inconsistency. However, this issue has been resolved in the past by relying on several iterations and by setting an adapted periodical maintenance rate. In other words, OLSR does not converge in a single iteration. OLSR may even not be fully operational for large and dense networks.

We consider convergence as the number of physical steps needed to make the protocol end. Still, we must distinguish here logical and physical steps. In order to elect a MPR node, it usually takes 2 logical steps, recursively performed until all two hops neighbors are covered. The physical steps are MPR's ability to notify the elected MPR nodes of their election. Indeed, in a perfect environment, MPR converges after successfully having notified all its MPR nodes of their respective elections. OLSR converges when all TC messages containing the MPR Selectors have been spread on the entire network. While it is hard to quantify the latter case, simulations we performed for the implementation of a modified OLSR including mobility predictions showed us that the former case is very critical. As a matter of fact, in our implementation, we tried to limit the logical steps for MPR election; in other words, we only allowed one single logical step. Nodes gather information on their neighbors, perform the two steps of the original heuristic of MPR, and notify their respective MPR nodes. In a perfect environment, all this should be far enough to make the protocol converge. Yet, we noticed that packet losses and the order of packet receptions were simply skewing the whole process up.

Let us first consider the order of packet decoding. In OLSR RFC3626 [18], upon reception of a packet, a node first considers in that order, Asymmetric links, Symmetric links, MPR links and Lost links. A typical example of such decoding problem is as follows. Consider new two-hop neighbors with an asym status, while the one-hop relaying node's status is asym but could become sym, since its ID is in its neighbor's asym link. However, this message will be ignored depending on the decoding order within the asym status. Indeed, while considering the asym link of its new two hop neighbors, the status of the relaying node is still, to the node's point of view, an asym link thus not qualified to receive two hop neighbors. But if it considers first its one hop neighbor's status, it updates it to sym and then adds freely the two hops neighbors to the list. We can find several other message discarding problems that are connected to the message decoding order, either within similar or different statuses. Yet, this is more an implementation issue than a MPR misconception. Yet, several implementations of OLSR ignore this problem and rely on multiple retransmissions to correct this issue. Consequently, several physical iterations are needed for each node to elect the correct MPRs and reach optimality.

Figure 1, we depict these convergence issues. We simulated the OLSR protocol using the Naval Research Laboratory OLSR implementation. As we want to show convergence issues, we simulated OLSR on a static network without traffic. We are convinced that nodes mobility and traffic will even worsen our results. Nevertheless, in the final version of the paper, we will include traffic and mobility in order to prove our conviction.

On Figure 1(a), we see that MPR needs on average 3 seconds to converge. In other word, during this time interval, non-stable MPRs are elected. The convergence time is here defined as the time before a node obtains symmetric links of all of its neighbors. We also see on the same figure that this spread itself to MPR convergence, or the time from which MPR computed optimal MPRs. No stable and optimal MPRs are obtained before 4 seconds on average. Therefore, OLSR cannot expect to create stable routes during this time interval. As we previously mentioned iterations, we show in Figure 1(b) the average number of iterations before MPR converges. We see that MPR needs on average 7 iterations before being able to provide OLSR with accurate topological data. These observations are important since they were obtained on a static network. If we consider mobility, every time the topology is changed, OLSR looses between 3 to 4 seconds before being able to reorganize its routes[2]. Moreover, these results also show that in highly mobile or heavy loaded networks, OLSR never completely converge as one might expect.



(a) Average time for OLSR convergence

(b) Number of iteration for convergence

Figure 1: Illustration of the convergence issue in OLSR

Another, serious issue that cannot be improved by a particular implementation is network inconsistency due to message losses. We should consider here two kinds of message losses for MPR. In order of importance: *messages containing links physical status*, and *messages containing links logical status*. While the former naturally represents the channel status, the latter is what we call Very Important Packets (VIPs).

Actually, the **weakest link** in OLSR comes from the **strongest link** of MPR. As we mentioned in previous sections, what is making MPR so optimal for flooding is its selective retransmission, also called **MPR flooding**. However, this is a very critical issue since perfect flooding for MPR and efficient routes for OLSR highly depends on this particular feature. If non accurate MPR selectors are obtained and propagated due to VIP losses, consequences arise at each step of OLSR. MPR nodes will not be coherent with the topology, MPR flooding stops being efficient and OLSR establishes suboptimal routes. In simulation we performed, we encountered serious network inconsistencies due to those VIP losses. Yet, as mentioned before, by setting the proper maintenance rate, this convergence issue is

---

[2] We do not consider OLSR own link state convergence time which also increases this time

simply reduced to a waste of resource but actually does never jeopardize OLSR protocol. Still, the objective in research in wireless ad hoc networking is precisely to improve resource management.

7 Conclusion and Clues for Improving OLSR

In this paper we presented OLSR requirements and MPR properties and showed the particular interactions between both protocols that make them hard to separate. Basically, they have been designed to work together and have been tuned for strongest links of one to compensate the weakest links of the other. We specifically exhibited the intrinsic properties of MPR that makes it so efficient but also depicted its major drawbacks. We finally mentioned that tuning of configuration parameters was a key aspect of the optimality of OLSR. To summary, any change of MPR heuristic, flooding control protocol, addition of a CDS protocol, or modifications of the proposed configuration parameters in the OLSR RFC3626 leads to performance decreases.

We also depicted in this paper the convergence issues OLSR experiences through MPR. We could observe that MPR needs on average 3 seconds to obtain symmetric links of all its neighbors, and cannot compute stable MPR before 4 seconds. In number of iterations, this mean that MPR needs at least 7 iterations before being able to provide OLSR with MPR selectors. This is a key issue that makes even more sense for mobile and heavy loaded networks, where the network is constantly reorganization and where transmission reliability is decreased. In the final version of this paper, we will provide similar results but including mobility and traffic.

In order to really improve OLSR, the basic property that should be investigated is **OLSR's weakest link**. OLSR is basically doing a great job in route optimization. It only needs correct and accurate data, with an efficient flooding for its TC messages. Therefore, investigations should be performed at the MPR level. As stated before, we should not try to find another flooding control protocol, but keep MPR and improve it. We should definitively try to remove MPR's requirement to notify its neighbors of their MPR elections.

Basically, nodes should be able to elect themselves as MPR. Wu et al. [19] developed a typical technique that they called *Extended Multipoint Relays (EMPR)*. Nodes use 3-hops neighborhood in order to accurately decide on their MPR status. This is a very promising protocol even if obtaining 3-hops knowledge is of $O(9 \cdot n)$, and is subject to packet losses and chain reaction of inconsistency. However, this drawback is compensated by nodes' ability to view the true connectivity needed to obtain a reduced number of MPR.

Nevertheless, this is not the main feature we are interested in here. Nodes ability to self-elect as MPR is the feature we seek in order to improve MPR. Unfortunately, as mentioned by Jacquet [20], the self-extraction capability of MPR status is not of a significant help since it is similar to CDS flooding. Consequently, we should extend Wu's work to not only allow nodes to elect themselves as MPR, but also to know **who needs them as a relay**. This could be called *Reverse Extended Multipoint Relays*. This has, to the best of our knowledge, never been investigated, and we basically do not know if it is feasible. Still, we think that such protocol would be a key improvement for OLSR, since it would less rely on mutual communications. This would thus reduce network inconsistency, yet with still the same flooding efficiency and route optimality, however at a much lower cost in term of resources.

References

[1] Brad Williams and Tracy Camp, ”Comparison of broadcasting techniques for mobile ad hoc networks”, in *Proc. of the 3rd ACM international symposium on Mobile ad hoc networking & computing (MobiHoc’02)*, pp. 194-205, 2002, Switzerland.

[2] A. Laouiti *et al*, ”Multipoint Relaying: An Efficient Technique for Flooding in Mobile Wireless Networks”, *35th Annual Hawaii International Conference on System Sciences (HICSS’2001)*, Hawaii, USA, 2001.

[3] T. Clausen *et al*, ”Optimized Link State Routing Protocol”, *IEEE INMIC*, Pakistan, 2001.

[4] A. Amis *et al*, ”MaxMin D-Cluster Formation in Wireless Ad-Hoc Networks”, In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, March 1999.

[5] M.R. Garey and D.S. Johnson, ”Computers and intractability. A guide to the theory of NP-completeness”, Freeman, Oxford, UK, 1979.

[6] C. Adjih, Ph. Jacquet, and Laurent Viennot, ”Computing connected dominated sets with multipoint relays”, in *INRIA Rapport de Recherche No. 4597*, INRIA Rocquencourt, France, 2002.

[7] R. G. Gallager, P. A. Humblet, and P. M. Spira, ”A distributed algorithm for minimum-weight spanning trees”, *ACM Trans. on Prog. Lang. and Systems*, Vol. 5, No. 1, pp. 66-77, Jan. 1983.

[8] N. Li, J.C. Hou, L. Sha, ”Design and analysis of an MST-based topology control algorithm”, in *Proc. of the IEEE Infocom Conference*, pp. 1702-1712, Vol. 3, San Francisco, 2003.

[9] F.J. Ovalle-Martinez *et al.*, ” Finding minimum transmission radii and constructing minimal spanning trees in ad hoc and sensor networks”, in *Journal of Parallel and Distributed Computing*, Vol. 65, No. 2, 132-141, February 2005.

[10] A. Busson, N.Mitton, and E. Fleury, ”An analysis of the Multi-Point Relays selection in OLSR”, in *INRIA Rapport de Recherche No. 4568*, INRIA Rhône-Aples, France, 2005.

[11] V. Bharghavan, and B.Das, ”Routing in Ad Hoc Networks Using Minimum Connected Dominating Sets”, *International Conference on Communication (ICC’97)*, Montreal, Canada, June 1997.

[12] B. Das, R. Sivakumar, and V. Bharghavan, ”Routing in Ad Hoc Networks Using a Spine”, *International Conference on Computer and Communications Networks (ICCN’97)*, Las Vegas, USA, September 1997.

[13] R. Sivakumar, B. Das, and V. Bharghavan, ”An Improved Spine-based Infrastructure for Routing in Ad Hoc Networks ”, *International Conference on Communication (ICC’97)*, Athene, Greece, June 1998.

[14] I, Stojmenovic, M. Seddigh, J. Zunic, ”Dominating Sets and Neighbors Elimination-based Broadcasting algorithms in Wireless Networks”, in *Proc. IEEE Hawaii Int. Conf. on System Sciences,* January 2001.

[15] J. Wu and H.L. Li, ”On Calculating Connected Dominating Set for Efficient Routing in Ad Hoc Wireless Networks”, in *Proc. of the* 3^{rd} *ACM Int. Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pp. 7-14, 1999.

[16] Emanuel Baccelli, Philippe Jacquet, ”Flooding Techniques in Mobile Ad Hoc Networks”, in *INRIA Rapport de Recherche No. 5002*, INRIA Rocquencourt, France, 2003.

[17] Navid Nikaein, Houda Labiod , and Christian Bonnet, ”Distributed Dynamic Routing Algorithm for Mobile Ad-Hoc Networks”, *Proc. MobiHOC 2000*, USA/Boston, 2000.

[18] T. Clausen and P. Jacquet, ”Optimized Link State Routing Protocol (OLSR)”, *www.ietf.org/rfc/rfc3626.txt*, Project Hipercom, INRIA, France, October 2003.

[19] Fei Dai and Jie Wu, ”An Extended Localized Algorithm for Connected Dominating Set Formation in Ad Hoc Networks”, in *IEEE Transaction on Parallel and Distributed Systems*, Vol. 15, No. 10, pp. 909-920, October 2004.

[20] Philippe Jacquet, ”Performance of Connected Dominating Set in OLSR protocol”, in *INRIA Rapport de Recherche No. 5098*, INRIA Rocquencourt, France, 2004.