World Applied Sciences Journal 21 (12): 1739-1747, 2013
ISSN 1818-4952

DOI: 10.5829/idosi.wasj.2013.21.12.154

A Hierarchical Version of OLSR for Manet

[1]A. Choukri, [1,2]A. Habbani and [1]M. Elkoutbi

[1]SIME Lab, MIS Team, ENSIAS, University of Mohammed V SOUISSI, Rabat, Morocco
[2]LEC Lab, MIS Team, EMI, University of Mohammed V AGDAL, Rabat, Morocco

Abstract: Due to the dynamic nature of Mobile Ad Hoc Network (MANET) the Quality of Service (QoS) requires several improvements. In this paper, we propose a novel hierarchical routing protocol based on clustering approach inspired on the heuristic Max-Min D-Cluster that we have improved and implemented on standard OLSR. In order to validate this new release named OLSR-MaxMin2C we studied its stability and we compared its performances to those of other alternatives. The metrics we used in evaluating the performance of networks were Average end-to-end Delay, Control Routing Overhead and Packet Delivery Ratio according to VBR traffic (H.264) in addition to CBR traffic. Experimental results show that the control traffic is highly reduced and the improvement in the network performances is gained.

Key words: MANET · QoS · OLSR and Clustering

INTRODUCTION

Mobile Ad-Hoc Network (MANET) is formed by mobile nodes connected using wireless links without any central infrastructure. Nodes are free to move arbitrarily; thus, the network topology may change randomly and dynamically.

Ad-Hoc Networks operates without using any such infrastructure. Each node moves randomly and may join or leave a network anytime during the lifetime of the network. Where a source node and a destination node are not within direct range, the intermediate nodes act as routers by using multi-hop schemes. Hence, the most challenging task is to provide quality-of-service (QoS) routing with several constraints such as unpredictable topology, limitation due to a nature of devices (limited memory size and computing capacity) and limitation in communication medium (limited bandwidth, interference, etc).

The aim of a routing algorithm is to provide a strategy that discovers the best route that links up two nodes in network. To meet this aim, the strategy should take into account the changes in network such as the unpredictable topology, the number of links, the power consumption and the bandwidth etc.

Many of the routing protocols for Ad-Hoc Networks are classified as either proactive or reactive routing protocols. Proactive routing protocols try to collect information about the MANET through proactive exchange of messages about their local topology. These protocols reach rapidly their limits when increasing density and mobility of nodes. However, reactive protocols find a route on demand by flooding the network with Route Request packets and require an important delay to find and to use the route that links up two nodes.

Consequently when increasing density of nodes, network generates more traffic control and requires more routing tables beyond the necessary. Therefore, any protocol family cannot perform well on Ad Hoc network with high density of nodes.

So as to solve this latest problem linked to the routing protocol with high density of nodes, some solutions have been proposed such as the hierarchical protocols. In this approach the close nodes are geographically grouped in a cluster. Moreover, this approach uses different routing schemes in the same cluster (intra-cluster) and between clusters (inter-cluster)

Hierarchical protocols allow to each node storing of all information of its cluster and only a part of information on other clusters. This approach has been proposed to reduce the size of routing tables in large networks.

We know that H.264 traffic is adopted by the multimedia application standards and it's adopted too by the industry consortium specifications. This extension can lead to broad heterogeneous applications especially when used with mobile ad hoc networks. On the other

Corresponding Author: A. Choukri, SIME Lab, MIS Team, ENSIAS, University of Mohammed V SOUISSI, Rabat, Morocco

hand, the MANET should improve their routing protocol in order to support the new multimedia applications such those use H.264 standard.

In this paper, we have improved the structure of OLSR, based on the clustering heuristic Max-Min D-Cluster [1] in order to improve the QoS in MANET in particular when considering multimedia traffic.

The rest of this paper is organized as follows: In the next section, we survey related work. The problem formulation is discussed in section 3, followed by the simulation environment used in this study in section 4. The results obtained in this simulation are also discussed in section 5. In the end, section 6 concludes the paper.

Related Work: In the previous works of Ad Hoc networks, several algorithms based on the clustering have been designed and proposed [2-4]. In this section, we describe the most used algorithms.

In [5], Ephremides and al. proposed one of the oldest used algorithm "Lowest ID algorithm" or "LCA". The authors in [6] have used the degree of nodes in their algorithm "Highest degree". According to the mobile environment, the cluster-heads degree change frequently in these algorithms. Therefore, the cluster-heads produced in these latest algorithms cannot play their role like cluster-heads for a long time.

The two algorithms described previously have been combined in a new algorithm named "CONID" in the paper [7]. This algorithm use the connectivity degree of nodes as primary key and the nodes ID as secondary key to select the cluster-heads.

The authors of [8] propose a novel algorithm named “Least Cluster Change” (LCC) that combines the “Lowest ID” and “Highest Degree” algorithms. Moreover, a maintenance step is added in order to minimize the cost of cluster-heads restructuring of network.

The Lowest Relative Mobility Clustering Algorithm (MOBIC) is proposed in [9]. Furthermore, the aim of this algorithm is to include the mobility metric to create the cluster-heads. Although this algorithm considers the mobility when restructuring the cluster-heads by reducing the number of their changing, the performances of this latest algorithm are less than the two described previously.

So as to select the cluster-heads, the Distributed Clustering Algorithm (DCA) [10] and Distributed Mobility Adaptive Clustering (DMAC) [11] algorithms combines several metrics in one weight which associated to each node.

The authors in [12] have proposed, an algorithm named 3hBAC (3-hop Between Adjacent Cluster-heads). However, the 3hBAC algorithm generates the disjoint clusters such as cluster-heads of two adjacent clusters are in three hops from one another. So as to maintain the clusters, the authors have combined the 3hBAC and the LCC algorithm in order to minimize the structure changes. However, this algorithm suffers from the congestion problem caused by the large number of cluster-head generated in the network.

A novel approach to generate k-clusters is proposed in [13]. However, this algorithm category generates k-clusters where any member in a cluster is at most k hops to its corresponding cluster-head.

In this paper, we propose an algorithm that combines the advantages of algorithms that generate disjoint clusters and those that generate k-clusters. Our algorithm is a lightweight in the sense that it is simple and without adding any control message to the network.

Problem Formulation: We know that the video codec have never stopped evolving in recent years such as MPEG-2, MPEG-4 and H.264. Accordingly, the theoretical H.264 compression ratio outperforms those of MPEG-2 and MPEG-4 which leads to improve the network transmission and can improves the viewing image quality especially in real time communication [14]. Due to these characteristics H.264 has already been adopted by the multimedia application standards and by the industry consortium specifications. This extension can lead to broad heterogeneous applications especially when used with mobile ad hoc networks. On the other hand, the MANET should improve their routing protocol in order to support the new multimedia applications such those use H. 264 traffic.

Based on previous researches, most of the applications generate variable bit rate (VBR) multimedia traffic inversely to the constant bit rate (CBR) traffic. Thus, the quantity of transmitted information in case of VBR traffic changes by time unit. Moreover, the variation degree depends on type and application behavior. Therefore, this variability decreases the QoS performances especially when considering the packet delivery ratio as shown in [15].

Most of routing protocol designed for Ad Hoc network with small and medium density of nodes and with low mobility provides best performances. However, due to the increasing nodes density [16] or to nodes agitation [17] the control traffic dominates the communications. Therefore, this problem leads to an increasing latency and routing overloaded. These problems can be explained by the process flooding to discover routes. Moreover, because nodes must rebroadcast the received packets the flood consumes the bandwidth. Furthermore, the packets distribution causes other problems such as collision and redundancy.

The objective of this work is to overcome the limitations mentioned previously. So, inspired from the heuristic Max-Min D-Cluster [1] that we have improved and adapted to the implementation on the standard OLSR, we obtained a new hierarchical protocol based on clustering approach named OLSR-MaxMin2C in order to improve the QoS in MANET in particular when considering multimedia traffic.

Thus, we have used in the experimental analysis, in addition to CBR traffic, VBR traffic that closely matches the statistical characteristics of a real trace of video frames generated by an H.264 encoder [11].

OLSR-MaxMin2C is based on the hierarchical architecture which considered one of used effective solution to broadcasting information in routing Ad Hoc network. Furthermore, this solution is interesting because it based on Connected Dominating Set (CDS) which used to optimize the communication costs. Although MANET has no fixed infrastructure, a virtual backbone [18] can be built by nodes in a CDS which its components are the only ones involved in information broadcast. Thus, more it is compact, more it will be effective.

This approach is based on the clustering concept that virtually cut out the network into groups of nodes geographically close called clusters. They are usually identified by a particular node called cluster-head. In most clustering algorithms, clusters are constructed from a particular metric which allows assigning a leader to each node, the cluster is then composed of cluster-head and all nodes attached to it (Fig 1).

Figure 1 shows a hierarchical structure with one level. There are also proposals for hierarchical structures multilevel in which clusters are then grouped into other clusters of higher levels and so on. Each level contains the cluster-heads and gateways of lower level (CDS).

The aim of this approach is to decrease the routing table size and its maintenance costs by involving only the nodes forming the backbone in the broadcasting. Therefore, this can lead to minimize the excessive redundancy emission.

Thereafter, we describe in detail our heuristic and we compare the new release of OLSR (OLSR-MaxMin2C) to standard OLSR and to sec-OLSR [19, 20] (hierarchical protocol based on the density of nodes to form clusters, created by MIS team of SIME laboratory, ENSIAS, M5S University, Rabat, Morocco) in term of Routing Control Overhead, Average End-to-End Delay and Packet Delivery Ratio (PDR).

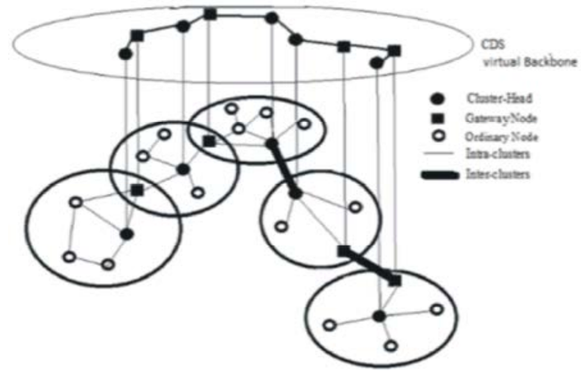

Fig. 1: Cluster Configuration

OLSR: OLSR is a proactive routing protocol designed for the Ad Hoc network. It is based on link state routing. By using the Multipoint Relays (MPR), OLSR minimizes the necessary traffic on each node in order to discover the topology of network [21]. The idea of multipoint relays is to minimize the flooding of broadcast packets in the network by reducing duplicate retransmissions in the same region. Each node *i* of the network independently selects a set of nodes in its one hop neighbors, which retransmits its packets. This set of selected neighbor nodes, called the multipoint relay of *i* and denoted MPR (*i*) is computed as follows: it is the smaller subset of one-hop neighbors with a symmetric link, such that all two-hop neighbors of i have symmetric links with MPR (*i*). This means that the multipoint relays cover (in terms of radio range) all the two-hop neighbors. The Figure 2 shows the multipoint relay selection by node *i*. Each node *i* maintains the set of its multipoint relay selectors (MPR selectors). This set contains the nodes that have been selected by *i* as a multipoint relay. Node *i* only forwards broadcast messages received from one of its MPR

Periodically, each node sends a HELLO message to all its neighbors. This message contains three lists of nodes representing the state of the link:

- List of neighbors addresses from which the node has received a HELLO packet;
- List of neighbors addresses which are accessible by a bidirectional link;
- List of neighbors addresses that the node has selected as multipoint relays.

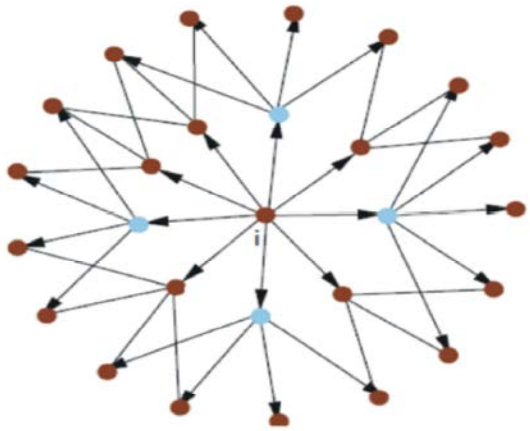

Fig. 2: Selected Multipoint relays of node i

Each node must detect the neighbor nodes with which it has a direct and bi-directional link. The uncertainties over radio propagation may make some links uni-directional. Consequently, all links must be checked in both directions in order to be considered valid. For this, each node periodically broadcasts its HELLO messages, containing the list of neighbors known to the node and their link status. The HELLO messages are received by all one-hop neighbors, but are not forwarded. They are broadcast at a low frequency determined by the refreshing period HELLO interval (the default value is 2 seconds) [22].

Clustering: Mobile Ad Hoc network can be designed such as a graph G= (V, E) where V designates the nodes and E represents the communication links. The clustering process is based on a virtual dividing of V into a set of geographically closed groups $\{V_1, V_2, \ldots, V_k\}$ such as (1):

$$V = \mathrm{U}_{i=1}^{k} Vi \tag{1}$$

These groups are named clusters and they are not necessarily disjointed. However, each cluster is identified by a particular node called cluster-head. So as to select a cluster-head a metric or a combination of metrics can be used such as: identifier, degree, density and mobility….Furthermore, according to the density and mobility of nodes the clustering algorithm efficiency is evaluated in term of the number clusters formed and in term of clusters stability.

Proposed Alternative Description: The initial idea of hierarchical routing is to allow each entity to store all the information of its cluster and only part of information on other clusters. This minimizes the routing table size and the generated traffic amount.

Clustering also has other advantages: It facilitates the sharing of resources and allows spatial reuse of radio frequencies to minimize interference [23]. More importantly, the organization of a network also provides greater stability [15].

Our alternative OLSR-MaxMin2C is based on the heuristic Max-Min D-Cluster [1] which generates disjoint clusters with D is a parameter that indicating cluster radius in number of hops. Therefore, we can tell that our algorithm generates the clusters which in same time disjoints and type k-clusters. As described, the link state information of two-hop neighbors of a node is represented by lists contained in HELLO messages. This information is broadcasted regularly within OLSR network. So as not to change the structure of control messages and not affect the operating mechanism of OLSR, we assigned the value 2 to the parameter D. which is already used by the principle of MPRs for flood information. Therefore, with this new value we can generate clusters of two hops, disjointed and balanced. For this reason we have named this release OLSR-MaxMin2C. Moreover, cluster-heads with OLSR-MaxMin2C are calculated based on the IDs of nodes within one and two hops neighbors. Knowing that, ID of symmetric neighbors two-hop are counted from the control messages of OLSR protocol as described in RFC 3626 [14].

Heuristic: OLSR-MaxMin2C produces clusters whose radius is fixed at 2 (It was assigned the value 2 at parameter D) on the basis of the nodes identifiers. Cluster-head is the node with the smallest identifier among the biggest identifiers in a neighborhood of two hops.

The Heuristic Runs in Two Phases:

- Phase recuperating the largest identifier in the neighborhood at 2 hops (2-neighborhood);
- Phase recuperating the smallest identifier among the largest in the 2-neighborhood.

The election of cluster-head is not based neither on the largest identifier nor the smallest, but the smallest among the greatest.

The purpose of this technique is to ensure that a node elected cluster-head does not keep its status permanently (counterbalance of the system)

Since the information of the 2-neighborhood is available for each node (OLSR principle), these two steps are made locally. Thus, even with the two phases, OLSR-MaxMin2C is therefore inexpensive in terms of messages and latency

Below We Present the Heuristic in Detail:

- Every node broadcasts the WINNER value which represents his identifier to its 2-neighbors and assign to this value the biggest ID heard among its 2-neighbors (2).

$$MAX\_ID_u = \max_{v \in N^{d}(u)} ID_v \tag{2}$$

- Every node broadcasts this new value WINNER to its 2-neighbors (Floodmax procedure) and collects those sent by its 2-neighbors.
- Every node keeps the smallest ID (WINNER) among those received during the previous step (smallest ID among the biggest ID broadcasted) (3).

$$MIN\_ID_u = \min_{v \in N^{d}(u)} MAX\_ID_v \tag{3}$$

- Every node broadcasts the MIN_IDu to 2 hops and collects those sent by his 2-neighbors.
- Choice of cluster-head:
  - If u sees his identifier among the MIN_ID of its neighbors, it is selected as a cluster-head.
  - Else, if u has seen the identifier of one or several nodes among the MAX_ID and the MIN_ID at the same time of his neighbors, it elects the one that have the lowest ID.
    - Else, U Elects the Biggest Identifier in its 2-neighborhood

Simulation Environment: To begin the experimental study, we validate the stability of OLSR-MinMax2C in term of number of generated clusters and average lifetime duration of cluster. Furthermore, we evaluate the same protocol in term of performances routing control overhead, average end-to-end delay and packet delivery ratio based on CBR and VBR (H.264) traffic. Finally we compare our protocol to sec-OLSR and to standard OLSR. Simulations have been carried out by Network Simulator 2.34 NS-2. In Table 1, we provide all simulation parameters.

Table 1. Simulation Parameters

Parameter	Value
Routing Protocols	Standard OLSR, sec-OLSR, OLSR-MaxMin2C
Simulation Time	1200 s
Number of nodes	10, 20, 30, 40, 50, 60, 70, 80, 90, 100
Environment Size	1000 m x 1000 m
Traffic Type	CBR, VBR (H.264)
Maximum Speeds	10 m/s
Mobility Model	Random Waypoint

Performance Metrics: About the simulation results, we have used the routing control overhead, average end-to-end delay, packet delivery ratio, node eccentricity and cluster diameter as metrics in order to evaluate the performances of the protocols described:

- Routing control overhead: is measured as the average number of control packets transmitted at each node during the simulation. Each hop is counted as one separate transmission.
- Average end-to-end delay: The delay of a packet is the time it takes the packet to achieve the destination after it leaves the source. The average packet delay for a network is obtained by averaging over all packets and all source destination pairs. The average end-to-end delay T_{Avg} is calculated as showing in equation (4):

$$T_{AVG} = \frac{\sum_{i=1}^{Nr}\left(H_r^i - H_t^i\right)}{Nr} \tag{4}$$

H_t^i : Emission instant of package I, H_r^i reception instant of package i, N_r the total number of packets received

- Packet Delivery Ratio: Ratio of the data packets successfully delivered to the destination.
- Eccentricity: We used e(u/C) to denote the eccentricity of node u in a cluster C. Where the eccentricity of a node is the biggest distance between u and any other node in the same cluster C:

$$e(u/C) = \max_{v \in C(u)}(d(u, v)). \tag{5}$$

- Diameter : The diameter of a cluster C, denoted by D(C) is the largest eccentricity in this cluster:

$$D(C) = \max_{u \in C}(e(u/C)). \tag{6}$$

RESULTS DISCUSSION

In this section we present the stability validation of OLSR-MaxMin2C. Thereafter, we present the simulation results of its validation and his comparison to sec-OLSR and to standard OLSR.

Satability of the Algorithm: As we know, the congestion problem is proportional to the number of clusters generated [24]. This stability depends also on minimum lifetime of a cluster before restarting the clustering process.

A node in MANET with OLSR routing receives HELLO messages regularly from his neighbors. Since we are based on those messages to form clusters, a node may change his state on each message reception. This can lead to instability in clustering system [19]. To solve this problem, a node should keep his state for reasonable period [20]. For this reason, we increase the period between broadcasts HELLO messages to three times. This refresh period represents the minimum lifetime of a cluster.

Simulation Results on the Clusters Characteristics of OLSR-MaxMin2C and sec-OLSR are as follows:

Table 2: Clusters Characteristics

	D(C)	*e(u/C)*
OLSR-MaxMin2C	3.8	3.6
Sec-OLSR	3.3	3.0

Based on described results on Table 2, we note that a node is more eccentric within its cluster ie that each formed cluster by the algorithm contains a reasonable number of nodes. Which implies that the algorithm builds less clusters. Therefore, we conclude that the structure of our proposition gives good results in term of stability which remains to be confirmed by the simulations related to the number of generated clusters and their lifetimes.

Number of Generated Clusters: As described previously, generating of a great number of clusters causes the congestion problem [24]. As solution we have increased the interval between two HELLO messages to three times more.

Figure 3 shows the number of clusters generated by OLSR-MaxMin2C depending on the node density when the period between sending two messages HELLO equal to that of standard OLSR and when we increase it to three times.

We observe that the generated clusters with three times interval of HELLO messages is reduced significantly

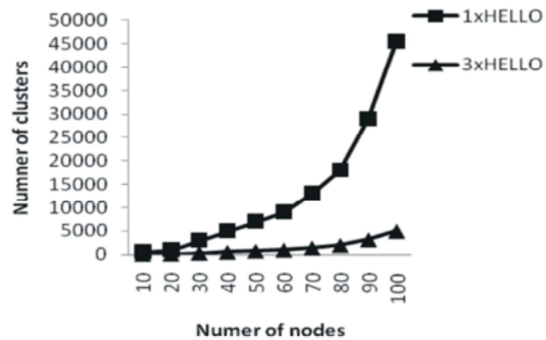

Fig. 3: Clusters Number vs Network density

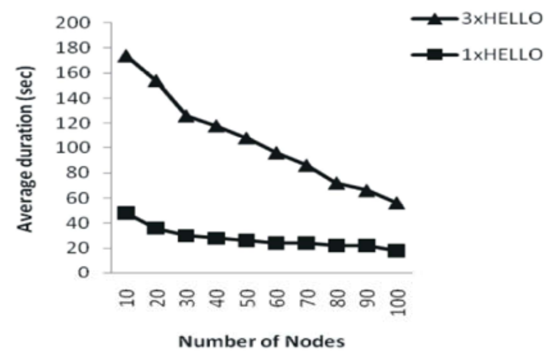

Fig. 4 : Average duration of a cluster vs. Network density

than those generated in normal case. From this result, we can conclude that when we force nodes to retain their status for a minimum of three times more than the normal we can solve the congestion problem and improve network performances

Cluster Average Life Time: Due to topology dynamic in Ad Hoc network the state of each node can drastically change. This will have a negative impact on the algorithm stability.

The Figure 4 shows the behavior of the cluster average duration according to density of nodes when the period between sending two HELLO messages equal to that of standard OLSR and when we multiply it three times.

As showing, by expanding the refreshing period of HELLO message interval to three times more than the normal we have increased remarkably the cluster average lifetime in network. Therefore, the cluster formation process becomes stable.

Routing Control Overhead: The Figure 5 shows the routing control overhead of standard OLSR, sec-OLSR and OLSR-MaxMin2C according to density of nodes. According to the same figure (Fig 5) OLSR-MaxMin2C reduces significantly the quantity of control messages broadcasted especially when increasing density of nodes.

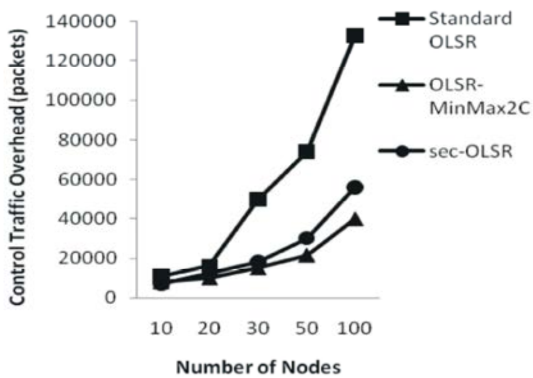

Fig. 5: Routing Control Overhead vs Number of nodes

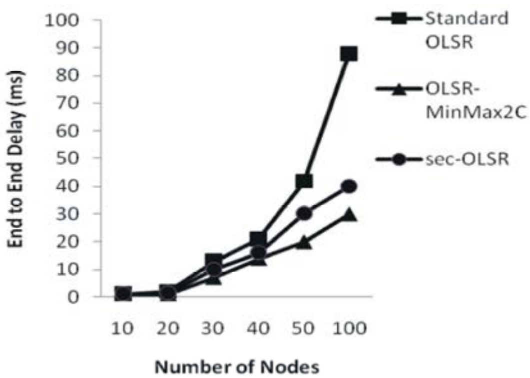

Fig. 6: End to End Delay vs number of nodes (CBR Traffic)

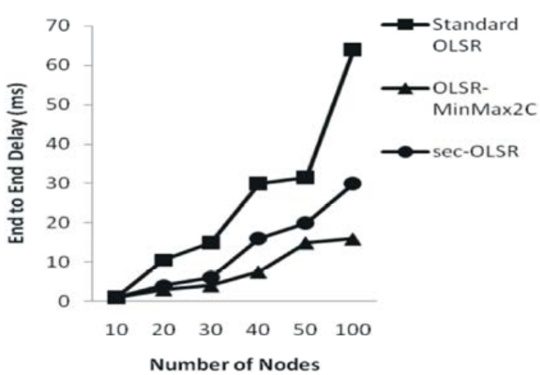

Fig. 7: End to End Delay vs number of nodes (VBR Traffic)

Therefore, this result leads to gain more bandwidth which can be exploited to improve other performances such as the throughput.

This result can be interpreted by the structure of our algorithm whose the behavior is: OLSR-MaxMin2C behaves like standard OLSR in inta-cluster, on the contrary it involves in inter-cluster only the nodes which forms the backbone.

Average End-to-End Delay

CBR Traffic: The aim of the Figure 6 is to show the efficiency of OLSR-MaxMin2C in term of average end-to-end delay. Moreover, it presents the delay of standard OLSR, sec-OLSR and OLSR-MaxMin2C according to density of nodes. As showing, the OLSR-MaxMin2C outperforms other in term of delay while increasing density of nodes.

The latest result can be explained by the fact that the communication between two remote clusters is done only through the nodes that forms the backbone. Consequently, this optimizes automatically the path in term of hop number in order to reduce the end-to-end delay.

VBR traffic (H.264): It is important to verify the efficiency of OLSR-MaxMin2C within conditions which illustrate reality of final applications such as using of multimedia traffic.

The Figure 7 shows the efficiency of OLSR-MaxMin2C in term of average end-to-end delay which achieved by using H.264 multiservice traffic. Moreover, it presents the delay of standard OLSR, sec-OLSR and OLSR-MaxMin2C according to density of nodes.

As showing, the OLSR-Maxmin2C takes less time to deliver data to the destination than other.

To conclude, according to the achieved results OLSR-MaxMin2C is efficient with the real time applications sensitive to delay.

Packet Delivery Ratio

CBR Traffic: In order to evaluate the OLSR-MaxMin2C we have compared this latest to standard OLSR and sec-OLSR in term of packet loss.

According to the Figure 8, OLSR-MaxMin2C delivers packets to the destination more than sec-OLSR and standard OLSR.

To conclude, based on the reduced control traffic quantity the OLSR-MaxMin2C improves the quantity of data traffic by choosing optimal paths according to number of hops.

VBR traffic (H.264): We also evaluated OLSR-MaxMin2C by comparing it with standard OLSR and sec-OLSR in term of packet loss based on the same H.264 variable bite rate traffic.

According to the Figure 9, OLSR-MaxMin2C is efficient in terms of loss especially for variable bit rate traffic.

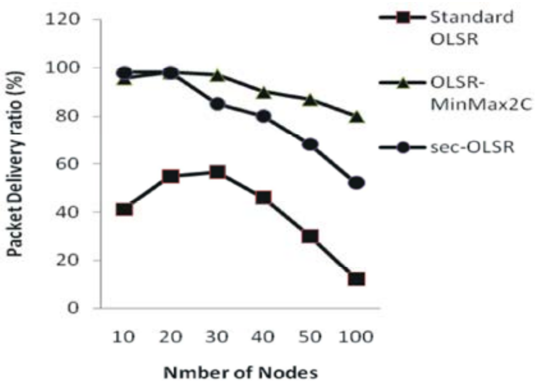

Fig. 8: Packet Delivery Ratio vs Number of nodes (CBR Traffic)

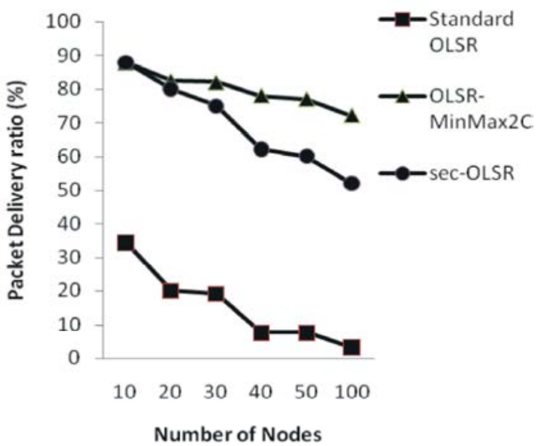

Fig. 9: Packet Delivery Ratio vs Number of nodes (VBR Traffic)

Finally, we can tell that our proposition can be used with multimedia applications which not tolerate packet loss.

GENERAL CONCLUSION AND PERSPECTIVES

In this article we addressed the limitations of traditional routing protocols in MANET and the need to involve the hierarchical protocols.

We have proposed a heuristic based on the notion of clustering that we have adapted and implemented on standard OLSR. The result of this implementation is a hierarchical routing protocol that we named OLSR-MaxMin2C (New OLSR version).

OLSR-MaxMin2C divides the network into disjoint clusters. It behaves like standard OLSR in intra-cluster and involves only nodes which form the Connected Dominating Set in inter-cluster. Thus it significantly reduces the amount of control traffic.

We compared OLSR-MaxMin2C to other alternatives and showed that it outperforms them in term of average end-to-end delay, control routing overhead and packet delivery ratio especially for multimedia traffic.

The followings are possible directions for future work. It would be interesting to consider a metric that combines mobility, density and power consumption to elect cluster-heads.

REFERENCES

1. Amis, A.D., R. Prakash, T.H.P. Vuong and D.T. Huynh, 2000. Max-Min D-Cluster Formation in Wireless Ad Hoc Networks. In Proceedings of IEEE Conference on Computer Communications (INFOCOM’00), 1: 32-41. Anchorage, Alaska,
2. Ismail Ghazi Shayeb, 2011. AbdelRahman Hamza Hussein and Ayman Bassam Nasoura. A Survey of Clustering Schemes for Mobile Ad-Hoc Network (MANET). American Journal of Scientific Research ISSN 1450-223X Issue, 20: 135-151.
3. Vivek Katiyar, Narottam Chand and Surender SoniA Survey on Clustering Algorithms for Heterogeneous Wireless Sensor Networks. Int. J. Advanced Networking and Applications. 02(04): 745-754 .
4. Abolfazl Akbari, Neda Beikmahdavi, Ali khosrozadeh, Omid Panah, Mehdi Yadollahi and Seyed Vahid Jalali, 2010. “A Survey Cluster-Based and Cellular Approach to Fault Detection and Recovery in Wireless Sensor Networks”, World Applied Sciences Journal, 8(1): 76-85.
5. Ephremides, A., J.Wieselthier and D. Baker, 1987. A design concept for reliable mobile radio networks with frequency hoping signaling. In IEEE, 75: 56-73.
6. Gerla, M. and J. Tsai, 1995. Multicluster, mobile, multimedia radio network. ACM/Baltzer Journal of Wireless Networks, 1(3): 255-265.
7. Chen, G. and I. Stojmenovic, 1999. Clustering and routing in mobile wireless networks. Technical Report TR-99-05, University of Ottawa,
8. Chang, C.C., H.K. Wu, W. Liu and M. Gerla, 1997. Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel. In Proceedings of IEEE Singapore International Conference on Networks (SICON’97), pp: 197-212. Singapore,
9. Basu, P., N. Khan and T.D.C. Little, 2001. A mobility Based Metric for Clustering in Mobile and Ad Hoc Networks. In Proceedings of International Conference of Distributed Computing Systems Workshop (ICDCSW’01), pp: 413-418. Phoenix, Arizona, USA,

10. Basagni. S., 1999. Distributed clustering for ad hoc networks. In Proceedings of Fourth International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN’99), pp: 310-315. Perth/Fremantle, WA, Australia,
11. Basagni, S., 1999. Distributed and mobility-adaptive clustering for multimedia support in multihopwireless networks. In Proceeding of Vehicular Technology Conference (VTF’99), 2: 889-893.
12. Yu, J.Y. and P.H.J. Chong, 2003. 3hBAC: a novel non-overlapping clustering algorithm for mobile ad hoc networks. In IEEE Pacific Rim Conference on Communications, Computers and signal Processing (PACRIM’03), 1: 318-321. Victoria, Canada,
13. Fernandess, Y. and D. Malkhi, 2002. K-clustering in wireless ad hoc networks. In Proceedings of the second ACM International Workshop on Principles of Mobile Computing, pp: 31-37. Toulouse, France, October 2002. ACM Press.
14. Choukri, A., M. Amnai, A. Habbani and M. Elkoutbi, 2012. AODV Protocol Behavior Based on H.264 Multiservice Traffic, International Review on Computers and Software (I.RE.CO.S.), September 2012 (Vol. 7 N. 5) - Papers Part A
15. Youssef Fakhri, Mohamed Amnai and Jaafar Abouchabaka, 2011. "Traffic Multiservice Based Performance Analysis of OLSR and AODV Protocols for Mobile Ad Hoc Networks", IJACT : International Journal of Advancements in Computing Technology, 3(8): 304-313.
16. Santivanez, C., B. McDonald, I. Stavrakakis and R. Ramanathan, 2002. On the scalability of ad hoc routing protocols. In Proceedings of Infocom,
17. Kwar, B.J., N.O. Song and L. Miller, 2004. On the scalability of ad hoc networks. In Proceedings of the IEEE Communication Letters,
18. Cheng, X., M. Ding, D.H. Du and X. Jia, 2006. Virtual backbone construction in multihop ad hoc wireless networks. International Journal of Wireless Communications and Mobile Computing, 6(2): 183-190.
19. Hajami, A. K. Oudidi and M.Elkoutbi 2010. ‘’An enhanced algorithm for MANET clustering based on Multi hops and Network Density’’ In proceedings of New Technologies of Distributed Systems (NOTERE), 2010 10th (NOTERE). 31 May - 2 July 2010, Tozeur - Tunisia
20. Hajami, A., K. Oudidi and M. Elkoutbi, 2010. “A Distributed Key Management Scheme based on Multi Hop Clustering Algorithm for MANET“, (IJCSNS) International Journal of Computer Science and Network Security, 10(1): 39-49.
21. Qayyum, A., L. Viennot and A. Laouiti, 2000. Multipoint Relaying : An Efficient Technique for flooding in Mobile Wireless Networks. Technical Report Research Report RR-3898, INRIA,
22. Jacquet, P., P. Mühlethaler, T. Clausen, A. Laouiti, A. Qayyum and L. Viennot, 2001. Optimized link state routing protocol for ad hoc networks. In Proceedings of the 5th IEEE Multi Topic Conference (INMIC 2001),
23. Clausen, T. and P. Jaquet, 2003. RFC 3626 - Optimized Link State Routing Protocol.
24. Rohit Khandekar, Guy Kortsarz and Vahab Mirrokni, 2012. Advantage of overlapping clusters for minimizing Conductance, LATIN'12 Proceedings of the 10th Latin American international conference on Theoretical Informatics, pp: 494-505. Springer-Verlag Berlin, Heidelberg.