SCALABLE MODEL FOR THE SIMULATION OF OLSR AND FAST-OLSR PROTOCOLS

Hakim Badis[1,2] and Khaldoun Al Agha[1,2]
[1]LRI Laboratory, University of Paris XI, Orsay, France
[2]INRIA Laboratory, Rocquencourt, France
{badis,alagha}@lri.fr

ABSTRACT

Technological advances and rapid development of the IEEE 802.11 standard have facilitated the growth of wireless local area networks (WLAN) and mobile computing in public domains. So, the routing protocols must support a very large and fast mobility of nodes over a very large ad-hoc network. In this paper, we present a scalable simulation model and results for the Optimized Link State protocol (OLSR) and the Fast-OLSR extension. OLSR is a proactive protocol, thus it periodically sends control packets to build and update the topology. Fast-OLSR extension is designed to meet the need for fast mobility in Mobile Ad-hoc NETworks (MANETs). The aim of this article is to evaluate the performance of Fast-OLSR in a very large ad-hoc networks by applying an extensible simulation model close to a real ad-hoc network. The simulation results were obtained with an IEEE 802.11 medium access control and physical layer model. Results show that the loss rate can be minimized for the Ad-hoc network of nodes with fast mobility by implementing OLSR and Fast-OLSR protocols.

I. INTRODUCTION

Growing interest has been given to the area of Mobile ad-hoc networking since the apparition of powerful radio devices allowing the connection of mobile nodes. A mobile ad-hoc network (MANET) [1] is a collection of nodes, which are able to connect on a wireless medium forming an arbitrary and dynamic network with *wireless links*. Implicit in this definition of a network is the fact that links, due to node mobility and other factors, may appear and disappear at any time. This in a MANET implies that the topology may be dynamic and that routing of traffic through a multi-hop path is necessary if all nodes are to be able to communicate.

A key issue in MANET is the necessity that the routing protocols have to respond rapidly to topological changes in the networks. At the same time, due to the limited bandwidth available through mobile radio interfaces, it is imperative that the amount of control traffic, generated by the routing protocols is kept at a minimum. Different routing protocols are proposed in the MANET divided into the following categories: proactive, ractive and hybride protocols.

OLSR [2,3] is a proactive, link-state routing protocol, employing periodic message exchange to update topological information in each node in the network. Topological information is flooded to all nodes, providing routes immediately available when needed. However, when a node is moving fast, its neighbors are not stable and change quickly. The OLSR protocol cannot detect the broken links quickly. So, the packets transmitted on an invalid link are lost. In order to minimize packet loss, an extension of the Optimized Link State Routing protocol (OLSR), denoted Fast-OLSR [4] is proposed. The nodes in a moving fast mode activate the fast moving mode by applying Fast-OLSR protocol. In this paper, we evaluate the performance of a large Ad-hoc network of nodes implementing OLSR and Fast-OLSR protocols.

This paper is organized as follows: in section II, we provide a brief overview of the IEEE 802.11 access technique and briefly describe the simulation model. In section III, we describe the three families of routing protocols (i.e., reactive, proactive and hybrid protocols) discussed in the MANET working group. In section IV, we briefly present OLSR, a proactive protocol suitable for dense and large mobile ad hoc networks. In section V, we show the Fast-OLSR extension that takes into account fast mobility. In section VI, we evaluate the performance of the Fast-OLSR extension on the basis of the simulation results. We present the simulation model that we have developed to represent a mobile network.

II. THE IEEE 802.11 STANDARD AND THE SIMULATION MODEL

A. IEEE 802.11 physical layer

We use the IEEE 802.11 [5] direct sequence (DS) system. The physical layer can offer a throughput of 1 or 2 *Mbps*, which takes into account the exact protocol overhead. We have used the following assumption: the broadcast packets and the point to point packets are sent at 1 Mbps. Our simulation model takes into account the exact overhead caused by the physical layer of IEEE 802.11 standard. For further detail refer to [5,6].

B. The IEEE 802.11 MAC scheme

With radio signals, it is not possible to directly detect collisions in a radio network. Indeed, it is not possible to listen to alien transmission while actually transmitting. Packet collisions must therefore be detected by another means. The IEEE 802.11 standard uses an acknowledgement for a point-to-point packet, broadcast packets are not acknowledged. The receiver sends this acknowledgement packet just after reception of the packet. The MAC scheme of the IEEE 802.11 is primarily based on a CSMA (Carrier Sense Multiple Access) scheme. The main principle of this access technique is a preventive listening of the channel to be sure that no other transmission is on the way before transmitting its packet. If the sensing of the channel indicates an ongoing transmission, then the node waiting to start its transmission draws a random backoff delay. At the end of the outgoing transmission, this backoff will be decremented whenever the channel is free (no carrier sensed). The node starts its transmission when its backoff delay reaches 0. This mechanism is presented in figure 1.

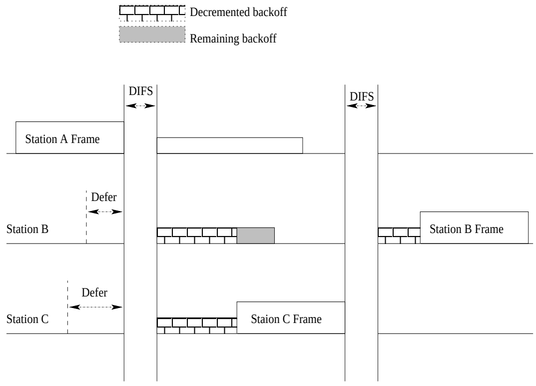

Fig. 1. IEEE 802.11 backoff mechanism

C. **The simulation model**

We use a simulation model near as that used in [6].

1) **Physical layer model***:* The main assumption of our physical layer model is that we have a linear superposition of signals sent by potential transmitters. This model naturally leads to the introduction of the signal strength sent by node j to node i denoted $\mathrm{CS}_{i,j}$.

The signal strength $\mathrm{Pow}(i)$ received by node i is therefore $\mathrm{Pow}(i) = \sum_{j=1}^{N} a_j \mathrm{CS}_{i,j}$ where $a_j = 1$ if node j is transmitting, or $a_j = 0$ otherwise. Simple propagation laws of radio signals usually have the following expression $\mathrm{CS}_{i,j} = \frac{P_j}{r_{i,j}^{\alpha}}$ where:

- P_j denotes the power sent by node j;
- $r_{i,j}$ denotes the distance between node i and node j and α denotes the signal decay, usually $2 \leq \alpha \leq 6$.

It should be noted that the only important assumption is the linearity of the model. We can actually use this linear model with all existing propagation models or precomputed figures. We need to introduce the carrier sensing parameter. This parameter is a threshold above which the channel is assumed to be busy. In a CSMA protocol, this threshold makes it possible to decide whether the channel is idle or busy. We will call this parameter the carriersenselevel. We need then to precise conditions to ensure the correct reception of packets. We will assume that a packet sent by node j to node i in the transmission interval $[t_b, t_e]$ is correctly received by node i if :

- $\forall t \in [t_b, t_e]$ $\mathrm{CS}_{i,j}(t) \geq$ *data-level.*
- $\forall t \in [t_b, t_e]$ $\frac{\mathrm{CS}_{i,j}}{\sum_{k \neq j} a_k(t) \mathrm{CS}_{i,k}(t)} \geq$ *capture-level.*

There are two parameters: the *data-level* and the *capture-level*. The *data-level* corresponds to the signal strength necessary to successfully transmit a signal. The *capture-level* corresponds to the minimum value of a signal to noise ratio to successfully decode a transmission.

2) **Medium access scheme simulation model:** This model is very close to real operations. However, it contains two simple approximations that simplify the acknowledgement and the RTS/CTS schemes. A complete description of this model can be found in [6].

III. **MANET ROUTING PROTOCOLS**

Ad-hoc networks are self-organizing, rapidly deployable, and require no fixed infrastructure. Nodes in an ad-hoc network may be highly mobile, or stationary, and may be very widely in terms of their capabilities and uses. The primary objectives of this new network architecture are to achieve increased flexibility, mobility and ease of management relative to infrastructured wireless networks. Ad-hoc network is itself mobile.

Several protocols exist, addressing the problem of routing in mobile ad-hoc networks. We can classify the routing protocols on the basis of their control behavior in the following categories: proactive, reactive and hybrid.

Proactive protocols use an adaptive system of routing based on periodic exchange of control messages. There may be various kinds of control messages: those which are sent locally (broadcast to one-hop) to enable a node to discover its local neighborhood; and those which are sent to be diffused in the network and which permit to distribute the topology information to all the nodes in the network. In a proactive approach, the routing protocol periodically updates the reach ability information in the nodes' routing table. Thereby a route is immediately available when needed. The cost for it is a use of substantial bandwidth for the periodic control traffic to acquire information, some of which may never be used. Proactive protocols include DSDV [7], OLSR [2,3] (an optimization of the link state algorithme OSPF [8])and TBRPF [9].

Reactive protocols do not take any initiative for finding a route to a destination, before the information is needed. The protocol attempts to discover routes only *on demand* by flooding its query in the network. During route discovery, the data packet is put on wait until the route becomes available. The drawback of this technique is that the braod consumption of the bandwidth for its global search (flooding) process, as well as adding large delays before sending data packet. Examples of reactive protocols include AODV [10] and DSR [11].

Hybrid protocols as ZRP [12] and CBR [13], employ a proactive scheme for one scope or type of requirements and functions as a reactive protocol for the other.

IV. **OPTIMIZED LINK STATE ROUTING PROTOCOL (OLSR)**

OLSR [2,3] is a proactive routing protocol, inherits the stability of a link state algorithm [14] and has the advantage of having the routes immediately available when needed due to its proactive nature. In a pure link state protocol, all the links with neighbor nodes are declared and are flooded in the whole network. The OLSR protocol is an optimization of the pure link state protocol for the mobile ad-hoc networks. First, it reduces the size of the control packets: instead of all links, it declares only a subset of links with its neighbors that are its multipoint relay selectors (see Section IV-A) [15]. Secondly, it minimizes the flooding of its control traffic by using only the selected nodes, called multipoint relays, to broadacst its messages. Therefore, only the multipoint relays of a node retransmit the packets. This technique significantly reduces the number of retransmissions in a flooding or broadcast procedure [16, 17]. OLSR protocol performs hop by hop routing, i.e. each node uses its most recent information to route a packet. Therefore, when a node is moving, its packets can be successfully delivered to it, if its speed is such that its movement could be followed in its neighborhood, at least.

A. Multipoint relay

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 *m* 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 (MPRs) of *m* and denoted MPR(m), is computed in the following manner: it is the smaller subset of one-hop neighbors with a symmetric link, such that all two-hop neighbors of *m* have symmetric links with $MPR(m)$. This means that the multipoint relays cover (in terms of radio range) all the two-hop neighbors. Figure 2 shows the multipoint relay selection by node *m*.

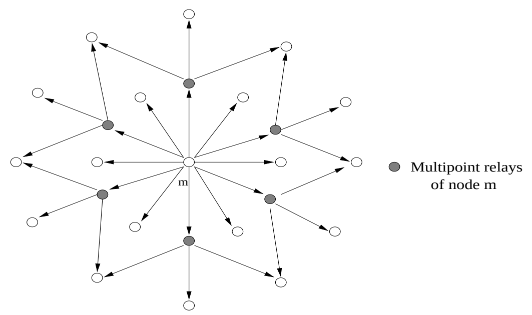

Fig. 2. Multipoint relays of node *m*

Each node *m* maintains the set of its *multipoint relay selectors* (MPR selectors). This set contains the nodes that have been selected by *m* as a multipoint relay. Node *m* only forwards broadcast messages received from one of its MPR selectors.

B. Neighbor sensing

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).

These *hello* messages permit each node to learn the knowledge of its neighbors up to two hops. On the basis of this information, each node performs the selection of its multipoint relays. These selected multipoint relays are indicated in the *hello* messages with link status MPR. On the reception of *hello* messages, each node can construct its MPR selectors table.

C. Topology information

Each node with a non-empty MPR selector set periodically generates a topology control message (*TC* message). This *TC* message is diffused to all nodes in the network at least every *TC_interval*. A *TC* message contains the list of neighbors that have selected the sender node as a multipoint relay. The information diffused in the network by these *TC* messages will help each node to build its topology table. Based on this information, the routing table is calculated. The route entries in the routing table are computed with *Dijkstra*'s shortest path algorithm [18]. Hence, they are optimal as concerns the number of hops.

The routing table is based on the information contained in the neighbor table and the topology table. Therefore, if any of these tables is changed, the routing table is re-calculated to update the route information about each known destination in the network.

V. FAST-OLSR

Fast-OLSR [4] protocol, an extension of OLSR, is designed to enable a fast moving node and keep the connectivity with other nodes in the network by quickly discovering a small number of neighbors and selecting among them a small number of multipoint relays. To do that, the Fast-OLSR uses a higher *hello* frequency to detect quickly its neighborhood changes and establishes a small number of symmetric links.
When a node detects that it is moving fast by a mechanism suggested in section 6.2.3, it activates the Fast-moving mode. In this mode, the node sends a *Fast-Hello* messages at a high frequency to establish Fast links. A *Fast-Hello* message has the same format with *hello* message, but its size is smaller because the node in fast moving must detect a reduced number of its neighbors and select over them its MPRs. Only nodes in Default Mode can be selected as MPRs. For further detail refer to [4].

VI. SIMULATION

In this section, we evaluate the performance of Fast-OLSR. We have carried out simulations to analyze OLSR and Fast-OLSR in different configurations and scenarios. We simulate OLSR and Fast-OLSR by OPNET simulator [19] as described in sections 4 and 5.

A. Simulation model

1) **Assumptions:** Our simulation model is based on the following main principles:

- The network is represented by a random graph model, in which nodes are placed randomly in a given region, initially;

- 80% of nodes implement OLSR and Fast-OLSR. 20% of nodes implement only OLSR protocol;
- All the nodes are identical (having the same capabilities) but they function independently of each other;
- If the nodes are mobile, each node independently decides its movement: its speed and the direction;
- The node can either receive or transmit at a time, e.g., it can not receive anything while it is transmitting;
- The nodes access the transmission channel by using CSMA/CA protocol (described in section II);
- There are no turns around time between transmitting and receiving: the nodes can switch over between transmit and receive mode instantly;
- Mobility is uncorrelated among the nodes of a network and links fail independently.

2) **Topology generation model :** We generate the topology of the network by randomly distributing (random graph) the nodes in a given region $(area\_max\_x, area\_max\_y)$. Each node is represented by a subqueue and placed in the region by randomly selecting its x and y co-ordinates. The number of nodes is given as network parameter, it can reach 100000 nodes. We select the type of mobile (implement OLSR and Fast-OLSR or only OLSR) by randomly distributing in a way that 80% of these mobiles implement OLSR and Fast-OLSR and 20% implement only OLSR. Initially, all the mobiles are in Default mode. Figure 3 shows the node's model representaion thus their positions via co-ordinates x and y.

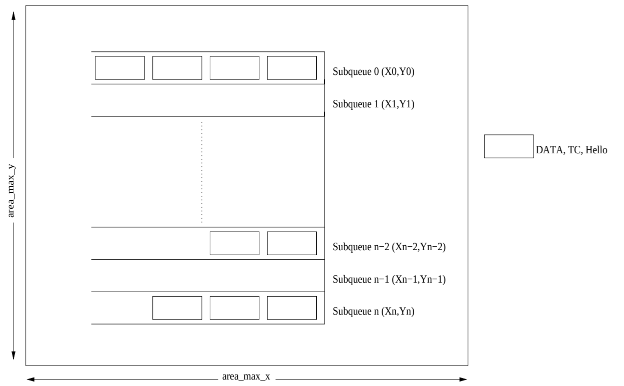

Fig. 3. Node's model

Each subqueue has a profile containing the identity of the corresponding node, the co-ordinates x and y, the maximum speed, the minimum speed and other useful informations for the implementation.

3) **Ad-Hoc Mobility Model :** The random ad-hoc mobility model [20] proposed in this section is a continuous-time stochastic process. Each node's movement consists of a sequence of random length intervals, during which a node moves in a constant direction at a constant speed. To calculate the co-ordinate of node n at t during an interval i of duration T_n^i, angle θ_n^i and speed V_n^i, we calculate at the first time the distance D covered by n, $D = V_n^i T_n^i$. Then, we calculate the x, y local co-ordinates, $x = D\sin(\theta_n^i)$, $y = D\cos(\theta_n^i)$. At the end, we calculate the global co-ordinates by changing scale. Figure 3 illustrates the movement of node n over six mobility intervals, each of which is characterized by its direction θ_n^i and distance.

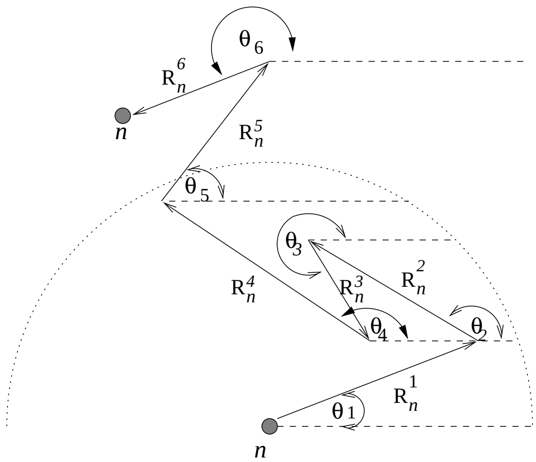

Fig. 4. Interval random mobility vectors

To obtain a balance between the arrivals and th departures in our area $(area\_max\_x, area\_max\_y)$, all the nodes leaving the zone of periphery are reinjected in the zone which is symmetrically opposed to them, thus eliminating the board's effects. figure 4 shows the reinjection of the nodes M1, M2 and M3.

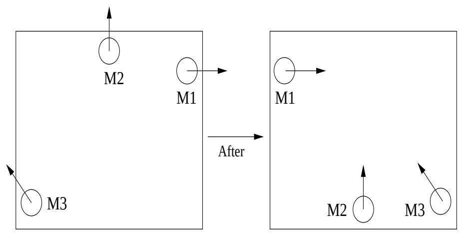

Fig. 5. Way-Round model

There are three important parameters: λ_n, $Speed\_Max$, $Speed\_Min$, for calculating the interval lengths, direction and speed. The interval lengths are exponentially distributed with mean $\frac{1}{\lambda_n}$. The direction of the mobile node during each interval is uniformly distributed over $(0, 2\Pi)$. The speed during each interval is uniformly distributed over $(Speed\_Min, Speed\_Max)$.

4) **The traffic and queuing model :** Data packets are generated at the nodes according to the Poisson distribution. The packet arrival rate at different nodes is independent from each other. In our simulation, all the nodes generate the same amount of load for the network. We have taken the mean packet size as $1Kbytes$. We limited the maximum packet size to $64Kbytes$. The control packets also follow this maximum size limitation.
Each node is a subqueue to queue up the new arriving packets. The new packets are queued up as long as there is a space in the subqueue. When the subqueue is full, the new packet is simply rejected. When a packet is successfully transmitted, the packet subqueue is freed from the transmit subqueue and that space is made available for a new packet. In our simulations, the destination for a data packet is randomly selected among all the destinations in the network, at each selection.

B. Implemented Algorithm

In this part, we show the OLSR and Fast-OLSR protocols implementation.

1) **Mobile in Default mode:** The implementation of OLSR protocol is represented in the figure 6. The source layer sends a DATA packet to the OLSR layer with a specified interarrival time. This DATA packet contains the address of the node, which has originally generated this message. This field should not be confused with the source address from the UDP header, which is changed each time to the address of the intermediate node witch is *re-transmitting* this message. It contains also the destination address and the packet length.

The OLSR layer includes the route in DATA packet. The next hop router is identified by the entry of the destination in the host routing table. The OLSR layer sends the *hello* (with the specified $Hello\_interval$), TC (with the specified $TC\_interval$)and DATA messages to the MAC layer.

The MAC layer transmits the packet by applying CSMA/CA protocol to its neighbor nodes and sends the packet well received to the OLSR layer. This layer uses an acknowledgement for a point-to-point packet, broadcast packets are not acknowledged. This acknowledgement packet is sent by the receiver just after reception of the packet.

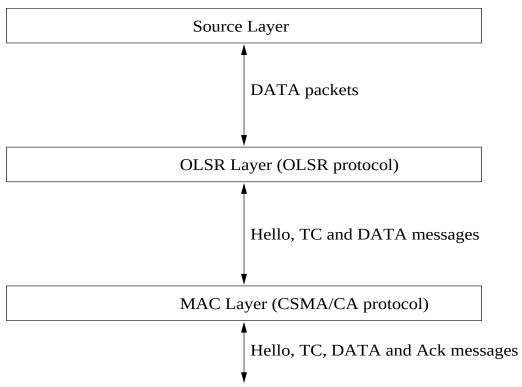

Fig. 6. The generic OLSR scheme in our simulation

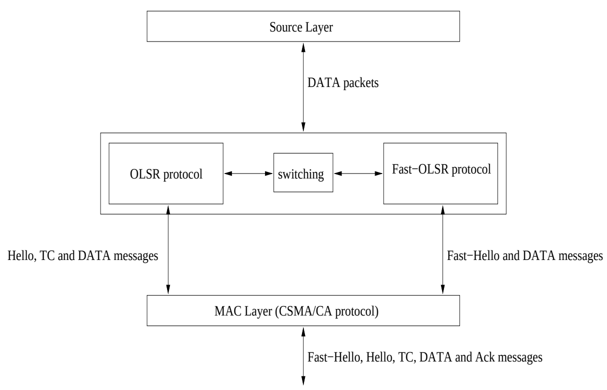

Fig. 7. The generic Fast-OLSR scheme in our simulation

2) **Mobile in Fast-Moving mode :** The implementation of Fast-OLSR protocol is represented in the Figure 7. Initially, all the mobiles are in default mode. When a node detects that it is moving fast, it switches to the Fast-Moving mode and starts sending *Fast-Hellos*. Also, when a node in Fast-Moving mode detects that it is no longer moving fast, it switches to the default mode. Fast-OLSR layer functions in the same way that OLSR concerning the DATA packets. It sends the *Fast-Hello* messages to the MAC layer with the specified $Fast\_hello\_interval$. There is no transmitting of TC messages because only nodes in Default Mode can be selected as MPRs. The switching module is described in the next paragraph.

3) **Switching to the Fast-Moving/Default Mode:** In order to detect a high or small number of changes in neighborhood, we proposed the model depicted in figure 8.

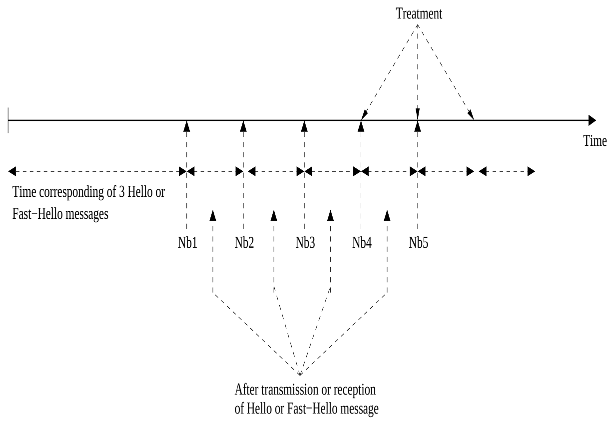

Fig. 8. The switching algorithm

A node *n* witch implement OLSR and Fast-OLSR must detect the number of changes in its neighborhood. Initially, this node waits a time corresponding of one-transmission and tow receptions of $Hello$ (if this node is in a default mode) or *Fast-Hello* (if this node is in Fast-moving mode) messages, or two transmissions and one reception of $Hello$ or *Fast-hello* messages, to record the number of new neighbors. This time is the smaller time to establish a symmetric links with other nodes. After this time, the node *n* records the number of changes in its neighborhood Nb_i. Henceforward, after each transmission or reception of $Hello$ or *Fast-Hello* message, the node *n* will record the number of changes (new neighbors, lost links) Nb_i. When the node *n* records Nb_i $(i \geq 4)$, if it is in the Default mode, it compares $(Nb_i - Nb_{i-3})/Nb_i$ with the given threshold T_1. if $(Nb_i - Nb_{i-3})/Nb_i \geq T_1$, the node *n* activates the mode Fast-moving. If the node *n* is in the Fast-Moving mode, it compares $(Nb_i - Nb_{i-3})/Nb_i$ with the given threshold T_2 $(T_2 < T_1)$. If $(Nb_i - Nb_{i-3})/Nb_i \leq T_2$, the node *n* activates the Default mode.

C. Advantages of the simulation model

The proposed simulation model is very extensible, there is several parameters: network parameters (number of nodes, region,...), OLSR parameters (Hello_interval, TC_interval, use of MPRs,...), Fast-OLSR parameters (Fast_hello_interval,threshold...), CSMA/CA parameters (Radio range, noise ratio, RTS/CTS, signal decay,...) and mobility parameters (Speed_min, Speed_max, inter_arrival,...). The number of nodes can reach 100000 nodes. With our method (each node is represented by a subqueue), the simulation model is very optimized that enables to reduce the machine time and consequently to increase the time of simulation. These, we can passe to a few weeks of simulation instead of few minutes that allows to refine the obtained results. The simulation model is very close to real Ad-hoc network operations. At each time, we can detect the position of mobiles by our mobility model.

D. **Evaluation results**

Figures 9 and 10 depict the loss of TC messages and rate versus the mobility. One curve (Default mode) is drawn with a network of 30 *nodes* in a region of $200^2 m^2$ where all the nodes implement only OLSR protocol. The speed is increased from 10 $meters/second$ $(36Km/hr)$ up to 60 $meters/second$ $(216Km/hr)$. We also keep the threshold T_1 as 0.8, T_2 as 0.2 and the *Fast_hello_interval* as 200 ms. Every 300 seconds, the node movement is re-determined. Each mobile node selects its speed and direction, which remains valid for next 300 seconds. We chose a signal decay $\alpha = 2$ and the transmission range as 50 *meters*. The second curve (With Fast-Moving) is drown with the same network of the previous curve but 80% of nodes implement OLSR and Fast-OLSR and 20% of nodes implement only OLSR protocol.

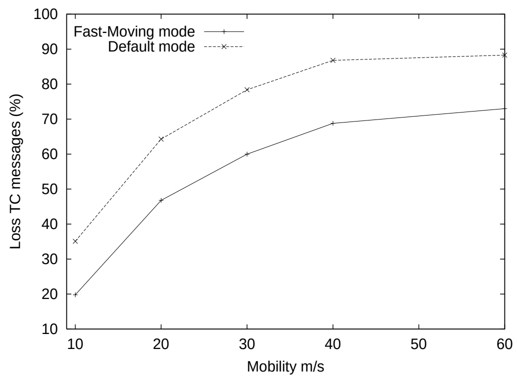

Fig. 9. Loss TC messages versus mobility

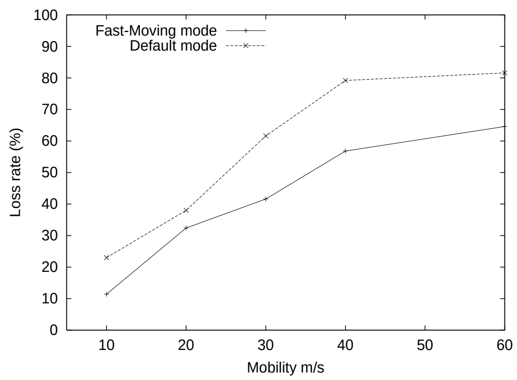

Fig. 10. Loss rate versus mobility

As figure 9 and 10 shown, the packet loss becomes greater as the speed increases and the loss of TC messages and rate in a network where all the nodes implement only OLSR protocol is greater than the loss of TC messages and rate in a network where 80% of nodes implement OLSR and Fast-OLSR and 20% of nodes implement only OLSR protocol. Indeed, the nodes lose the topology information of the network (Loss TC message). Consequently, it will have a great number of the unavailable routes. we note a variation from 13% ($10m/s$) to 20% ($40m/s$) in figure 9 and a variation from 12% ($10m/s$) to 20% ($40m/s$) in figure 10. With Fast-OLSR protocol, there are less of loss TC messages; therefore, the loss of DATA will be smaller.

Figures 11 and 12 depict the loss of TC messages and rate versus the mobility with a different *Fast_hello_interval*. For a speed of 40 $meters/s$ $(144Km/hr)$ and a loss rate less than 40%, the only possible *Fast_hello_interval* is 100*ms*. However, for a speed limited to 30 $meters/s$ $(108Km/hr)$ with a lost rate less than 40%, there are several possibilities for the *Fast_hello_interval*: 100*ms*, 200*ms*. We select the highest curve, which gives us a *Fast_hello_interval* of 200*ms*. So, for a given maximum acceptable loss rate and the maximum reachable speed, we can determine the largest *Fast_hello_interval*.

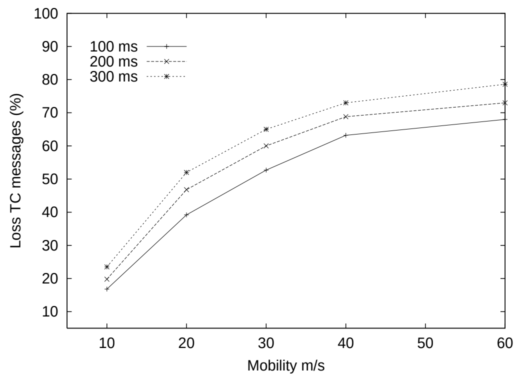

Fig. 11. Loss TC messages versus mobility

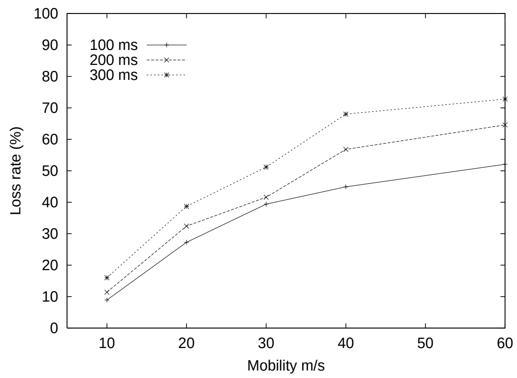

Fig. 12. Loss rate versus mobility

VII. **CONCLUSION**

In this work, we have studied the performance of the OLSR and the Fast-OLSR protocol (an extension of OLSR dealing with fast mobility). We have proposed a scalable simulation model close to real Ad-Hoc network. The number of nodes can reach 100000 nodes. This model is very extensible; we can easily extend the simulation's environment and give a very large simulation time. The mobility model is very complete and continuous-time, we can detect the position of mobiles at each time. We have also proposed a new model, which permits to each node the switch to the Fast-Moving/Default mode very quickly. The performances of a network with mobiles implemented OLSR and Fast-OLSR in terms of loss rate are performed compared to a network of mobiles implementing only OLSR protocol. Results show that the loss rate can be minimized for the Ad-hoc network of nodes with fast mobility by implementing OLSR and Fast-OLSR protocols.

REFERENCES

[1] J. Macker, S. Corson. Mobile Ad hoc Networking and the IETF. ACM Mobile Computing and Communications Review, 1998.

[2] A.Qayyum. Analyse and evalutaion of channel access schemes and routing protocols for wireless networks. Phd thesis, Computer Science Dept, Univ. Paris Sud in Orsay, Novembre 2000.

[3] P.Jacquet, P. Muhlethaler, A.Qayyum, A. Laoutiti, L.Viennot, T.Clausen. Optimized Link state Routing protocol (OLSR). Internet Draft, draft-ietf-manet-olsr-05.txt, 31 October 2001, Work in progress.

[4] M. Benzaid, P. Minet, K. A. Agha. Integrating fast mobility in the OLSR routing protocol. INRIA research reports, Juin 2002.

[5] IEEE 802.11 standard. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifi cations. June 1997.

[6] P. Muhlethaler, A. Najid. An effi cient Simulation Model for Wireless LANs Applied to the IEEE 802.11 Standard. INRIA research reports 4182, http://www.inria.fr, April 2001.

[7] C. E. Perkins and P. Bhagwat. Highly dynamic destination sequenced distance-vector routing (DSDV) for mobile computers, in SIG-COMM'91.

[8] J. Moy. OSPF version 2. Internet Standard, Request For Comments 2328, April 1998.

[9] Richerd G. Ogier, Fred L. Templin, Bhargav Bellur and Mark G.Lewis. Topology Broadcast based on Reverse-Path Forwarding (TBRPF). Internet Draft, draft-ietf-manet-tbrpf-03.txt, November 28 2001, Work in progress.

[10] C. E. Perkins, E. M. Royer and S. R. Das. Ad hoc on-demand distance vector (AODV) routing. Internet Draft, draft-ietf-manet-aodv-09.txt, November 2001, Work in progress.

[11] David B. Johnson, Yih-Chun Hu, David A. Maltz and Jorjeta G. Jetcheva. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR). Technical report, Rice University, AON Networks, Carnegie Mellon University, November 2001. draft-ietf-dsr-06.txt, Work in progress.

[12] Zygmunt J. Haas and Marc R. Pearlman. Zone Routing Protocol (ZRP) for AD HOC Networks. Internet Draft, draft-ietf-manet-zone-zrp-02.txt, June 1999, Work in progress.

[13] M. Jiang, J. Li, Y-C. Tay. Cluster Based Routing Protocol (CBRP). Internet Draft, draft-ietf-manet-cbrp-spec-01.txt, August 1999, Work in progress.

[14] E. Streenstrup. Routing in Communication Networks. Prentice Hall International, Inc., Englewood Cliffs, NJ, 1995.

[15] A. Qayyum, L. Viennot, A. Laouiti. Multipoint relaying: An effi cient technique for fboding in mobile wireless networks, Research Report RR-3898, INRIA, http://www.inria.fr, February 2000.

[16] P. jacquet, P. Muhletaler, A. Qayyum, A. Laouiti, T. Clausen, L. Viennot. Optimisation link state routing protocol. In IEEE INMIC, (Pakistan), December 2001.

[17] A. Qayyum, L. Viennot, A. Laouiti. Multipoint relaying technique for fboding broadcast message in mobile wireless networks. In HICSS: Hawai International Conference on System Sciences. Hawii, USA, January 2002.

[18] A. S. Tanenbaum. Computer Networks. Prentice Hall, 1996.

[19] OPNET simultor, http://www.opnet.com/.

[20] A. Bruce and T. Znati. A Mobility Based Framework for adaptive Clustering in Wireless Ad-Hoc Networks. Appears in IEEE journal on Selected Areas in Communications, Vol. 17, No. 8, August 1999.