ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

AODV CLUSTERING ALGORITHM BASED ON DENSITY AND NODES MOBILITY WITH A MOBILE REFERENCE

[1]M. SAADOUNE, [2]M. DYABI, [3]A. HAJAMI, [4]H. ALLALI

LAVETE Laboratory, Univ. HASSAN 1st, FSTS, Department of Mathematics and Computer Sciences, Morocco
E-mail: [1]saadoune.meryem@gmail.com, [2]mohamedyabi@gmail.com, [3]abdelmajidhajami@gmail.com, [4]hakim-allali@hotmail.fr

ABSTRACT

A mobile ad hoc network (MANET) is generally defined as a network that has many free nodes, often composed of mobile devices, thus the mobility of the nodes effects a much the performance of the network. AODV is a routing protocol for mobile ad hoc networks (MANET), it has low processing and memory overhead and low network utilization, and works well even in high mobility situation, but it has issues for large mobile networks.
A clustering architecture provides network scalability and fault tolerance, and results in more efficient use of network resources. For those reasons, we propose in this paper two kind of clustering algorithm in AODV: Density-based algorithm and mobility-based algorithm using a mobile reference. Our objective is to elect a reduced number of less mobile cluster heads.

Keywords: *Ad hoc, Mobility, Localization, Distance, Relative speed, AODV, Metric, Clustering.*

1. INTRODUCTION[1]

Mobile ad hoc network (MANET) is an appealing technology that has attracted lots of research efforts. Ad hoc networks are temporary networks with a dynamic topology which doesn't have any established infrastructure or centralized administration or standard support devices regularly available as conventional networks [1]. Mobile Ad Hoc Networks (MANETs) are a set of wireless mobile nodes that cooperatively form a network without infrastructure, those nodes can be computers or devices such as laptops, PDAs, mobile phones, pocket PC with wireless connectivity. The idea of forming a network without any existing infrastructure originates already from DARPA (Defense Advanced Research Projects Agency) packet radio network's days [2][3]. In general, an Ad hoc network is a network in which every node is potentially a router and every node is potentially mobile. The presence of wireless communication and mobility make an Ad hoc network unlike a traditional wired network and requires that the routing protocols used in an Ad hoc network be based on new and different principles. Routing protocols for traditional wired networks are designed to support tremendous numbers of nodes, but they assume that the relative position of the nodes will generally remain unchanged. In ad hoc, since the nodes are mobile, the network topology may change rapidly and unpredictably and the connectivity among the terminals may vary with time. However, since there is no fixed infrastructure in this network, each mobile node operates not only as a node but also as a router forwarding packets from one node to other mobile nodes in the network that are outside the range of the sender. Routing, as an act of transporting information from a source to a destination through intermediate nodes, is a fundamental issue for networks. [4]
The problem that arises in the context of ad hoc networks is an adaptation of the method of transport used with the large number of existing units in an environment characterized by modest computing capabilities and backup and fast topology changes.

According to the way of the creation and maintenance of roads in the routing of data, routing protocols can be separated into three categories, proactive, reactive and hybrid protocols. The pro-active protocols establish routes in advance based on the periodic exchange of the routing tables, while the reactive protocols seek routes to the request. A third approach, which combines the strengths of proactive and reactive schemes, is also presented. This is called a hybrid protocol.

Ad-hoc On-Demand Distance Vector routing protocol (AODV) [5] is a reactive routing protocol, who was standardized by the working group MANET [6] with IETF (Internet Engineering Task force), by the (RFC 3561). The protocol's algorithm creates routes between nodes only when the routes are requested by the source nodes, giving the network the flexibility to allow nodes to enter and leave the network at will. Routes remain active only as long as data packets are traveling along the paths from the source to the destination .When the source stops sending packets, the path will time out and close.
Clustering is the most popular method developed to provide resource management over mobile ad hoc networks. This technique is based on partitioning the network in smaller and manageable groups.

In this paper we propose a solution that enables each node in the network to determine the location of its neighbors in order to create a more stable and less mobile road. For that purpose, we locally quantify a metric collecting the distance and the value of motion (relative speed) to locally describe the mobility of the neighbors. Using this metric and the density of each node in the network we can select the less mobile cluster head.

The remainder of this paper is organized as follows. Section 2, describes briefly the AODV protocol. In Section 3, a summary of related work is presented. We present in Sections 4 and 5 how to quantify, evaluate, and estimate mobility in ad hoc network (Density and relative speed). Section 6 shows the algorithm used the quantification of our mobility's metric in AODV protocol. Section 7 shows our density-based clustering algorithm. Section 8 presents our mobility-based clustering algorithm with a mobile reference. Section 10 presents some simulations and results. Finally Section 11 concludes this paper.

2. AD HOC ON-DEMAND DISTANCE VECTOR

AODV is an on-demand protocol which is capable of providing unicast, multicast [7], broadcast communication and Quality of Service aspects (QoS) [8], [9]. It combines mechanisms of discovery and maintenance roads of DSR (RFC 4728) [10] involving the sequence number (for maintains the consistency of routing information) and the periodic updates of DSDV [11].
At the discovery of routes, AODV maintains on each node the transit information on the route discovery, the AODV routing tables contain (Table 1) :

- The destination address
- The next node
- The distance in number of nodes to traverse
- The sequence number of destination
- The expiry date of the entry of the table time.

Table 1: Route Request Contents

Route Request
Broadcast ID
IP source
Destination address
Hop number
Source Sequence number
Destination Sequence number

When a node receives a packet route discovery (RREQ), it also notes in its routing table information from the source node and the node that just sent him the package, so it will be able to retransmit the response packet (RREP). This means that the links are necessarily symmetrical (Figure 1). The destination sequence number field of a route discovery request is null if the source has never been linked to the destination, else it uses the last known sequence number. It also indicates in this query its own sequence number. When an application sends a route discovery, the source waits for a moment before rebroadcast its search query (RREQ) road, after a number of trials, it defines that the source is unreachable.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

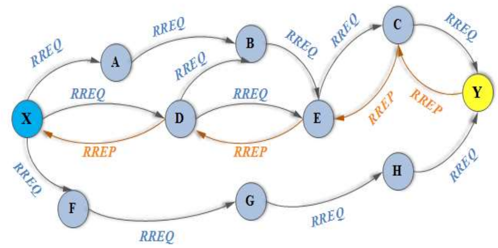

Figure 1: AODV route detection scheme possible between two nodes X and Y

Maintained roads is done by periodically sends short message application called "HELLO" , if three consecutive messages are not received from a neighbor, the link in question is deemed to have failed . When a link between two nodes of a routing path becomes faulty, the nodes broadcast packets to indicate that the link is no longer valid. Once the source is prevented, it can restart a process of route discovery.

AODV maintains its routing tables according to their use, a neighbor is considered active as long as the node delivers packets for a given destination, beyond a certain time without transmission destination, the neighbor is considered inactive. An entered routing table is considered active if at least one of the active neighbors using the path between source and destination through active routing table entries is called the active path. If a link failure is detected, all entries of the routing tables participating in the active path are removed.

3. Related Work

In [12], a geometric mobility metric has been proposed to capture and quantify the relative motion of nodes. The mobility measure between any pair of nodes is defined as their absolute relative speed taken as an average over time.

In order to arrive at the aggregate mobility metric for a particular scenario, the mobility measure is averaged over all pairs of nodes. This metric has certain deficiencies with respect to clustering: First, it assumes a GPS like scheme for calculation of relative speeds; in a MANET, especially in the indoors, we cannot assume the existence of GPS, and so we have to resort to other techniques for measuring relative mobility. Secondly, it is an “aggregate” mobility metric and does not characterize the local movement of the nodes in the neighborhood of a particular node, which is the primary reason for cluster head changes.

An AODV-based Clustering Approach for Efficient Routing in MANET [13]: In Clustering approach, the cluster head election is call upon for the constructing the path, reduce the communication over heads and scalability. For the path construction cluster-AODV routing protocol is applied and also the design goals of clustering algorithms are presented. To elect the cluster heads, this algorithm selects nodes having the weakest identifier which is just its IP address. But it isn’t because a node has a small identifier; it’s suitable to act as a cluster head.

A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks: This paper [14] presents a novel relative mobility metric for mobile ad hoc networks (MANETs). It is based on the ratio of power levels due to successive receptions at each node from its neighbors. They propose a distributed clustering algorithm, MOBIC, based on the use of this mobility metric for selection of cluster heads for selection of cluster heads. In this proposition nodes need to periodically exchange branch messages what’s lead to overload the network.

In the literature, several studies have addressed the problem of clustering in MANETs. To form clusters and elect cluster heads, each solution provides different criteria.

In [22], the authors propose a routing protocol based on clusters. To elect the cluster heads, this algorithm also selects nodes having the weakest identifier which is just its IP address. But it isn’t because a node has a small identifier; it’s suitable to act as a cluster head. They proposed a hierarchical OLSR version, a hierarchy according to nodes capabilities. The node capability depends on the amount and properties of its wireless interfaces. A node with several interfaces and large radio range will be selected as cluster head. If the network nodes have the same wireless interfaces properties, the routing finds the OLSR standard operation and there won’t be clustered structure. To form clusters, a new message called CIA (Cluster Id Announcement) is periodically sent by cluster heads to declare their leadership and invite

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

other nodes to join their clusters.

In [23], the authors propose a clustering mechanism for OLSR protocol. They introduce the forest and tree concept. The entire network is seen as a forest, where each cluster is considered like a tree and the branches represent the links between nodes. To select a root of the tree, the algorithm uses maximum local connectivity, i.e. nodes having more neighbors are designated as roots. In order to enable OLSR nodes to form and maintain trees, OLSR nodes need to periodically exchange branch messages (in addition to usual OLSR control messages).

In [24], authors use location information for cluster formation: the highest degree node in a neighborhood, i.e. the node with the largest number of neighbors is elected as CH. Experiments demonstrate that the system is not scalable: as the nodes number in a cluster is increased, a gradual degradation in the system performance is observed.

Moreover, in highly mobile environments, the re-affiliation rate increases due to node movements and as a result, the highest-degree node (the current CH) may fail to be re-elected even if it loses a single neighbor.

Our proposal presents a simple, light and quiet solution. First, our proposal doesn't add any new control message and the network isn't overloaded or slowed at all. No changes are made to standard control messages. Our solution works transparently with OLSR standard protocol without using GPS for node's location. Clusters are formed around the lowest mobile nodes, i.e., the node that has the less mobility value is elected as cluster head.

4. LOCAL QUANTIFICATION OF NEIGHBORING MOBILITY

In this section, we define how we estimate nodes mobility in ad hoc network. Mobility is quantified locally and independently of this localization of a given node. We represent this local quantification node mobility as the degree of spatial dependence of all nodes in the network.

a and b are two nodes that want to communicate in a MANET network.

Dab(t): the distance between nodes a and b at time t.
Va(b): the speed of b with respect to a.

$$Va(b) = \frac{dab(t\prime) - dab(t)}{t\prime - t} \qquad (1)$$

The interpretation of the value of this metric is done according to the sign of the latter.

If it is positive: Nodes move away from each other.

Else: Nodes move toward each other.

After the quantification of the speed between all nodes, we can describe the behavior of the node in the network by calculate the average of all speeds Avg(Vij).

If the average is very high we say that the network nodes are very agitated else the network is supposedly more stable.

The quantification of the distance can be done using 3 methods [15]: calculate the exact distance using GPS or using a distance calculation distance in a simulation environment, calculate the distance using the RSSI (Received Signal Strength Indication) and calculate distance using GPS-free.

Using one of methods above, every node can calculate the movement's speed of its neighbors.

By definition the relative speed is the variation in time of the distance between two mobiles.

5. ALGORITHM OF QUANTIFICATION DISTANCE IN THE AODV ROUTING PROTOCOL

In this part, we propose to use one of those methods in the first function of a AODV protocol (rout establishment between a source and a destination).

A node x wants to communicate with a node y.

x diffuse RREQ.

Each node receiving RREQ, calculates the distance between itself and the neighbor who sent him RREQ (in this part we use the exact distance or the distance using the Pr) and broadcasts its table [neighbors-distance-time] to its neighbors.

Each node calculates the relative speed between itself and its neighbors using the precedent formula. To use the third method for the quantification of the distance, the algorithm has to change.

A node x wants to communicate with a node y.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

x diffuse RREQ.

Each node receiving RREQ, calculates the distance between itself and the neighbor who sent him RREQ (in this part we use the exact distance or the distance using the Pr) [15], broadcasts its table [neighbors-distance-time] to its neighbors.

Each node calculates the relative speed between itself and its neighbors using the precedent distances.

We choose the reference that has the smallest value of speed and recalculate the newest distances using the third method in [15].

6. ALGORITHM OF QUANTIFICATION OF THE MOBILITY'S METRIC IN AODV ROUTING PROTOCOL

In this part, we propose to use one of those methods in the first function of a AODV protocol (rout establishment between a source and a destination).

A node b wants to communicate with a node a.

b diffuse E- RREQ (Table 2).

Each node receiving E-RREQ, calculates the distance between itself and the neighbor who sent him E-RREQ (in this part we use one of the methods listed in [15]) and broadcasts its table [neighbors-distance-time] to its neighbors (Figure 2).

Table 2: Enhanced Route Request Contents

E-Route Request
Broadcast ID
IP source
Destination address
Hop number
Source Sequence number
Destination Sequence number
** [neighbors-distance-time] **

Each node calculates the relative speed [16] between itself and its neighbors using the precedent formula.

In this part, we are sure that all nodes have all distances between themselves and their 2-hop neighbors.

All parts of the whole algorithm are repeated during the simulation.

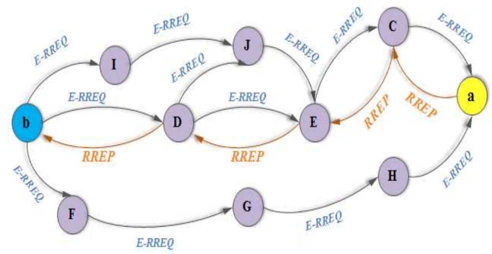

Figure 2: AODV route detection schema possible between two nodes b and a using E-RREQ

7. ALGORITHM OF QUANTIFICATION OF DENSITY'S METRIC IN AODV ROUTING PROTOCOL

Once RREQ is received, each node looks for the packet source node in its routing table.

If the node is not in the routing table then he has to insert it into the routing table as well as the neighbor list. Otherwise, it updates the expiration time in the routing table and the time to expiry of the list of neighbors.

The structure of the list of neighbors already exists (simple addition of time_expire fields)

The quantification of the density is through the route of the list of neighbors.

He was elected CH, each node having a density greater than that of its neighbors

Each node can make the decision to become cluster head or not locally as it can access the list of neighbors of all these neighbors.

8. AODV CLUSTERING ALGORITHM

In a clustered AODV network, each node can be in one of three states:

- State 0: not decided. When a node has just arrived, or it has just left its cluster and has no neighbors in its neighborhood, its status is not decided yet. There is no cluster head or cluster member. It must wait for the receipt of RREQ packet.
- State 1: Cluster head. The node was exchanged RREQ, and it has the highest metric value. It creates a cluster in which it was appointed head of the cluster.
- State 2: member. The node has exchanged RREQ; it has a low metric value compared to its symmetric neighbors, and is part of the cluster members.
  - ➢ Each node evaluates the mobility of the other nodes with which it

Journal of Theoretical and Applied Information Technology
30th April 2018. Vol.96. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

communicates.

- Each node receiving a RREQ, calculates the distance between itself and the neighbor who sent him RREQ Packet.
- Each node calculates the relative speed between itself and its neighbors using the precedent distances, and send those information [neighbors – Relative speed - time] into RREQ Packet.
- After receiving the RREQ, each node has a vision about the mobility of other nodes by computing the mobility average of each one.
- Upon receiving a RREQ, each node compares the neighbor's mobility with its own mobility to decide whether to become a cluster head or join the neighbor's cluster.
- Initially, each node begins with a status 0 (not decided). Upon receiving a RREQ, the node compares its own metric (Mi) with the metric of the packet it received (M).
- If (M > Mi), the node goes to state 1 (cluster head) because its metric value is lower than the metric of the received packet.

Once in state 1, node i triggers a counter Cptr. If after passing this timeout, the node i has received no RREQ, that means it has no neighbors in its radio range, so it decides to move to state 0 (not decided state).

- If (M < Mi), the node goes to state 2 (member) because its metric value is greater than that of the received packet.

In state 2, node i triggers a counter Cptr. If after passing this timeout, the node i has received no RREQ Packet, that means it has no neighbors in its radio range, so it decides to move to state 0 (not decided state).

- If the node i is in state 1 (respectively in state2), and it receives a RREQ Packet with (M > Mi) (respectively (M < Mi)), it remains in state 1 (respectively remains in state 2) because its state has not changed.
- If the node i is in state 1 (respectively in state2), and it receives a RREQ with (M < Mi) (respectively (M > Mi)), it moves to state 2 (respectively move to state 1) because its condition has to change.

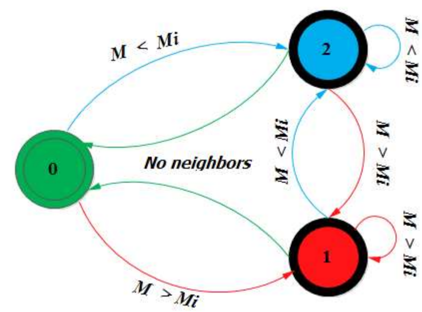

Figure 3: Clustering algorithm

9. MOBILITY MODELS

The performance of an ad hoc network protocol can change significantly when it's tested with different mobility models, but also when the same mobility model is used with different parameters. In addition, the choice of a model requires a model of data traffic that also influences the performance of the protocol. The performance of an ad hoc network protocol must be evaluated with the mobility model that is closest to the predicted real scenario, which may facilitate the improvement of the ad hoc network protocol. To evaluate the performance of our clustering algorithm, we performed simulations for five different types of mobility models described in the following paragraphs.

9.1. Random Walk: RW

A mobile node (MN), moves from its current location to a new location by randomly selecting a direction and a traveling speed.

The new speed and direction are selected from predefined ranges, [speedmin, speedmax] and [0.2π], respectively.

Each movement is in a constant time interval t or a distance d, at the end of which a new direction and a new velocity are calculated (Figure 4).

A memory less mobility pattern

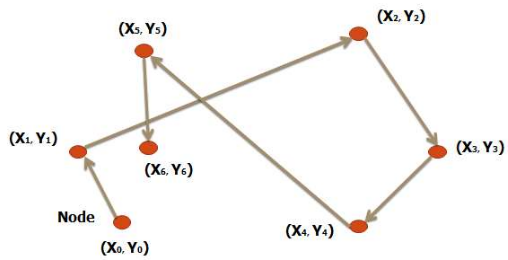

Figure 4: Node movement in Random Walk

9.2. Random Waypoint: RWP

Random Walk + a pause time

A mobile node begins by staying in a place for a certain period of time, once that period is over; it has performed the Random Walk (figure 5).

A memory less mobility pattern

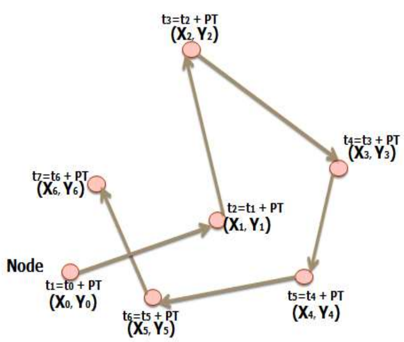

Figure 5: Node movement in Random Waypoint

9.3. Gauss-Markov

The Gauss Markov mobility model [18] is a memory model in the sense that the position and velocity of a node at any instant (t ') depend on the position and the velocity at time t; which creates a more flexible movement of the nodes. The position (x, y) of the mobile node and its speed v are updated each unit of time.

The speed $\boldsymbol{V_t}$ and direction $\boldsymbol{\theta_t}$ of a node is calculated as follows:

$$V_t = \alpha V_{t-1} + (1-\alpha)V_0 + \sqrt{(1-\alpha^2)}V_{t-1}^G \ldots\ldots\ldots\ldots(1)$$

$$\theta_t = \alpha\theta_{t-1} + (1-\alpha)\theta_0 + \sqrt{(1-\alpha^2)}\theta_{t-1}^G \ldots\ldots\ldots \quad (2)$$

The parameter $\boldsymbol{\alpha}$ $\boldsymbol{(0 \leq \alpha \leq 1)}$ is used to incorporate the degree of randomness while calculating the speed and direction of movement for a time period. The degree of randomness decreases as we increase the value of $\boldsymbol{\alpha}$ from 0 to 1. When $\boldsymbol{\alpha}$ is closer to 0, the degree of randomness is high, which may result in sharper turns. When $\boldsymbol{\alpha}$ is closer to 1, the speed and direction during the previous time period are given more importance (i.e., the model is more temporally dependent) and the node prefers to move in a speed and direction closer to what it has been using so far. Thus, the movement of a node gets more linear as the value of $\boldsymbol{\alpha}$ approaches unity.

The terms $\boldsymbol{V_{t-1}^G}$ and $\boldsymbol{\theta_{t-1}^G}$ are random variables chosen independently by each node from a Gaussian distribution with mean 0 and standard deviation 1.

If $(\boldsymbol{X_t}, \boldsymbol{Y_t})$ are the co-ordinates of a node during the beginning of time period t, then the coordinates $(\boldsymbol{X_{t+1}}, \boldsymbol{Y_{t+1}})$ of the node at the end of time period t (which is also the beginning of time period t+1), are given by equations (3) and (4) shown below.

The node thus moves from $(\boldsymbol{X_t}, \boldsymbol{Y_t})$ to $(\boldsymbol{X_{t+1}}, \boldsymbol{Y_{t+1}})$ during time period t with the $\boldsymbol{V_t}$ and in direction $\boldsymbol{\theta_t}$ determined from equations (1) and (2) respectively.

$$\boldsymbol{X_{t+1} = X_t + V_t * \cos(\theta_t) \ldots\ldots\ldots\ldots\ldots\ldots(2)}$$

$$\boldsymbol{Y_{t+1} = Y_t + V_t * \sin(\theta_t) \ldots\ldots\ldots\ldots\ldots\ldots(3)}$$

To ensure that a node does not remain near an edge of the simulation, the nodes are pushed away from the edge when they are within a certain distance from the edge.

9.4. Manhattan

As described in [19], it's an emulation of the model of circulation of mobile nodes in streets defined by maps.

A map is composed of a number of horizontal and vertical streets (Figure 6).

The mobile node is allowed to move along the grid of horizontal and vertical streets on the map.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

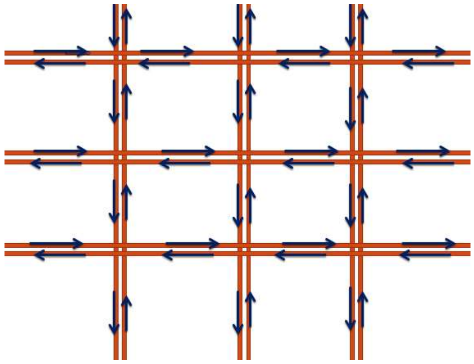

Figure 6: Manhattan Grid

The mobile node can turn left, right or go straight. This choice is probabilistic: the probability of passing in the same street is 0.5, the probability of turning to the left is 0.25 and the probability of turning right is 0.25.

9.5. Reference Point Group Mobility Model: RPGM

This model is described in [20][21]as another way to simulate the behavior of a group of nodes, where each node belongs to a group.

Use of a logical center (group leader), that determines the movement behavior of the group (Figure 7). Use the reference points that push them in the direction of the group.

The node can change its reference point.

The different nodes use their own mobility model and are then added to the reference point which drives them in the direction of the group. At each instant, every node has a speed and direction that is derived by randomly deviating from that of the group leader. This general description of group mobility can be used to create a variety of models for different kinds of mobility applications.

Group mobility as such can be used in military battlefield communications. One example of such mobility is that a number of soldiers may move together in a group. Another example is during disaster relief where various rescue crews (e.g., firemen, policemen, and medical assistants) form different groups and work cooperatively.

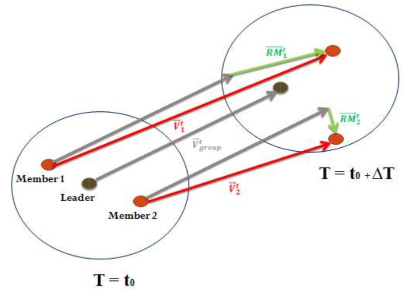

Figure 7: Node movement in RPGM

10. Simulations and Results:

In the following simulations, we applied our proposition to the AODV protocol .For this, we have been used the simulator NS-2 [17], with its implementation of AODV protocol of the version NS-2.35.

10.1 Environment

The network size considered for our simulations is (1000m×1000m) . The nodes have the same configuration, in particular TCP protocol for the transport layer and Telnet for the application layer. Time for each simulation is of 60s. For each simulation the mobility of the nodes is represented by the choice of a uniform speed between v_{min}= 0 and v_{max} = 40 m/s. The nodes are moved after a random choice of the new destination without leaving the network (1000m×1000m).

10.2 Discussions of results

We performed simulations using all mobility models cited before and we have recorded the average number of clusters built (which we notice NC) and the average time during which a cluster is maintained (C-Duration).

We used the two metrics, the first one based on Density and the second based on the mobility.

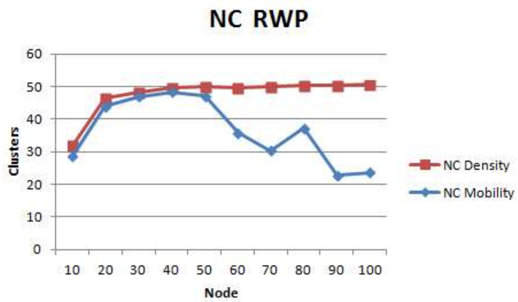

Figure 8: Average Number of Cluster = f (nbr nodes) In RWP V = 45m/s

Figure 8 shows the evolution of the number of clusters in relation to the number of nodes in a network using Random Way Point as a mobility model for a maximum speed of 45 m /s.

We notice that the number of clusters in the algorithm based on mobility is less than the algorithm based on density.

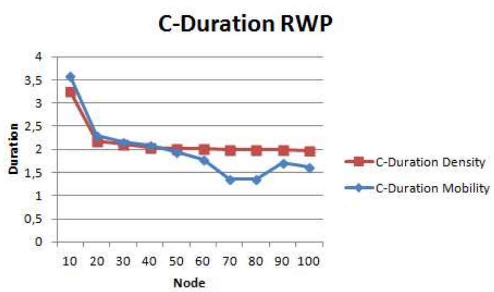

Figure 9: Average Cluster duration = f(nbr nodes) In RWP V = 45m/s

Figure 9 shows the behavior of the average time during which a cluster is built based on the number of nodes in a network using Random Way Point as a mobility model.

Contrary to what was expected, the duration of formation of cluster head becomes better in density-based algorithm once the node number exceeds 40 nodes despite the high number of clusters formed using this algorithm in the previous figure.

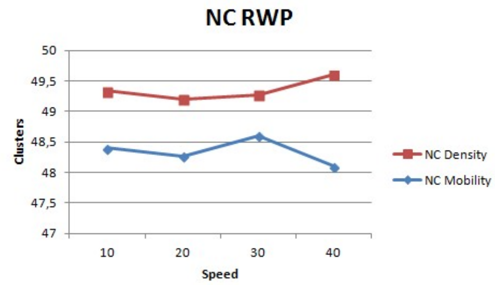

Figure 10: Average Number of Cluster = f (speed) In RWP

Figure 10 shows the evolution of the number of clusters in relation to the velocity of nodes in a network of 40 nodes using Random Way Point as a mobility model.

We notice that the number of clusters in the algorithm based on mobility is less than the algorithm based on density.

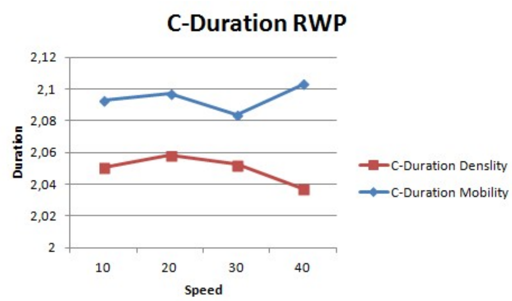

Figure 11: Average Cluster duration = f (speed) In RWP

Figure 11 shows the behavior of the average time during which a cluster is built based on the velocity of nodes in a network using Random Way Point as a mobility model.

We notice that the mobility-based algorithm has the best duration results compared to the density based algorithm, and it's what was expected given the previous figure.

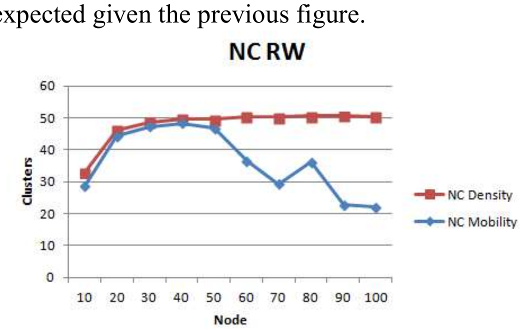

Figure 12: Average Number of Cluster = f (nbr nodes) In RW V = 45m/s

Figure 12 shows the evolution of the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

number of clusters in relation to the number of nodes in a network using Random Walk as a mobility model for a maximum speed of 45 m /s.

We notice that the number of clusters in the algorithm based on mobility is less than the algorithm based on density.

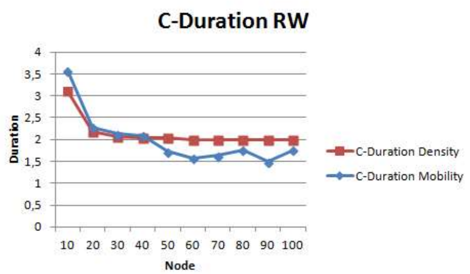

Figure 13: Average Cluster duration = f (nbr nodes) In RW V=45m/s

Figure 13 shows the behavior of the average time during which a cluster is built based on the number of nodes in a network using Random Walk as a mobility model.

Contrary to what was expected, the duration of formation of cluster head becomes better in density-based algorithm once the node number exceeds 40 nodes despite the high number of clusters formed using this algorithm in the previous figure.

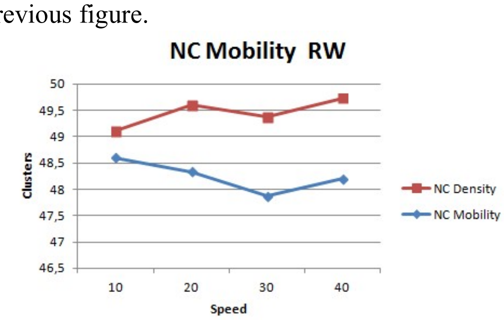

Figure 44: Average Number of Cluster = f (speed) In RW

Figure 14 shows the evolution of the number of clusters in relation to the velocity of nodes in a network of 40 nodes using Random Walk as a mobility model.

We notice that the number of clusters in the algorithm based on mobility is less than the algorithm based on density.

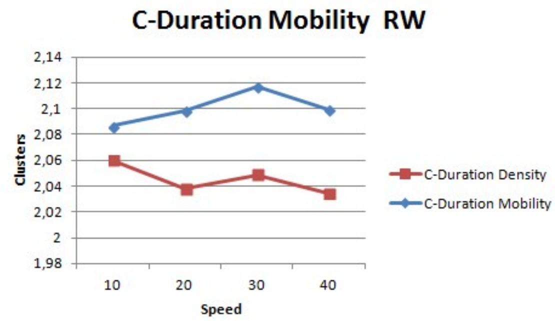

Figure 55: Average Cluster duration = f (speed) In RW

Figure 15 shows the behavior of the average time during which a cluster is built based on the velocity of nodes in a network using Random Walk as a mobility model.

We notice that the mobility-based algorithm has the best duration results compared to the density based algorithm, and it's what was expected given the previous figure.

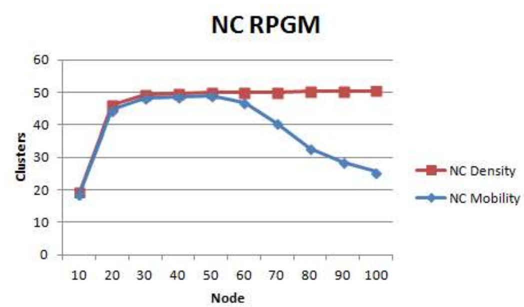

Figure 16: Average Number of Cluster = f (nbr nodes) In RPGM V=45m/s

Figure 16 shows the evolution of the number of clusters in relation to the number of nodes in a network using Reference Point Group Mobility as a mobility model for a maximum speed of 45 m /s.

We notice that the number of clusters in the algorithm based on mobility is less than the algorithm based on density especially when the number of nodes exceeds 50 nodes.

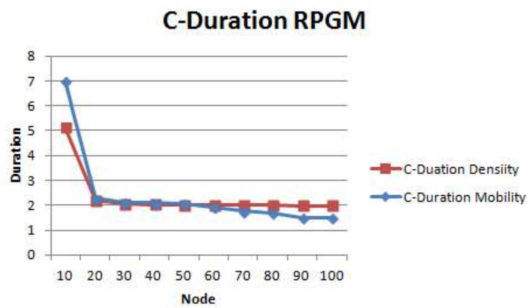

Figure 67: Average Cluster duration = f (nbr nodes) In RPGM V=45m/s

Figure 17 shows the behavior of the average time during which a cluster is built based on the number of nodes in a network using Reference Point Group Mobility as a mobility model.

Contrary to what was expected, the duration of formation of cluster head becomes better in density-based algorithm once the node number exceeds 60 nodes despite the high number of clusters formed using this algorithm in the previous figure.

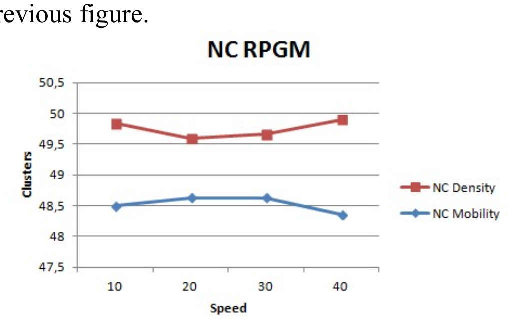

Figure 78: Average Number of Cluster = f (speed) In RPGM

Figure 18 shows the evolution of the number of clusters in relation to the velocity of nodes in a network of 40 nodes using Reference Point Group Mobility as a mobility model.

We notice that the number of clusters in the algorithm based on mobility is less than the algorithm based on density.

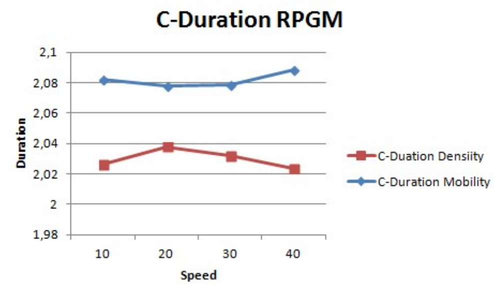

Figure 89: Average Cluster duration = f (speed) In RPGM

Figure 19 shows the behavior of the average time during which a cluster is built based on the velocity of nodes in a network using Reference Point Group Mobility as a mobility model.

We notice that the mobility-based algorithm has the best duration results compared to the density based algorithm, and it's what was expected given the previous figure.

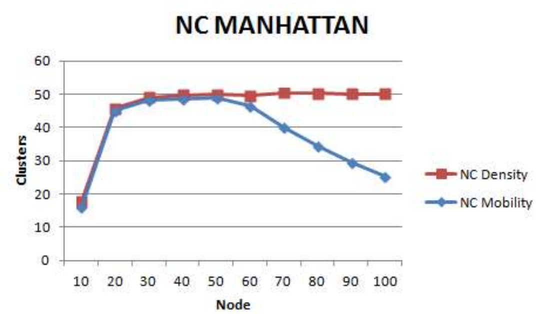

Figure 20: Average Number of Cluster = f (nbr nodes) In Manhattan V=45m/s

Figure 20 shows the evolution of the number of clusters in relation to the number of nodes in a network using Manhattan as a mobility model for a maximum speed of 45m /s.

We notice that the number of clusters in the algorithm based on mobility is less than the algorithm based on density especially when the number of nodes exceeds 50 nodes.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

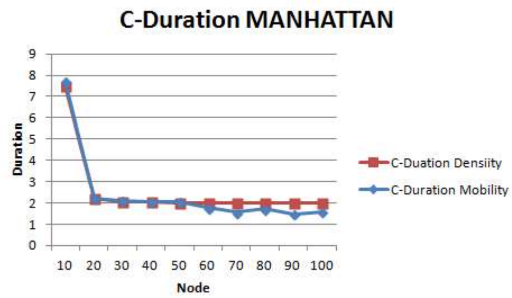

Figure 21: Average Cluster duration = f (nbr nodes) In Manhattan V=45m/s

Figure 21 shows the behavior of the average time during which a cluster is built based on the number of nodes in a network using Manhattan as a mobility model.

Contrary to what was expected, the duration of formation of cluster head becomes better in density-based algorithm once the node number exceeds 60 nodes despite the high number of clusters formed using this algorithm in the previous figure.

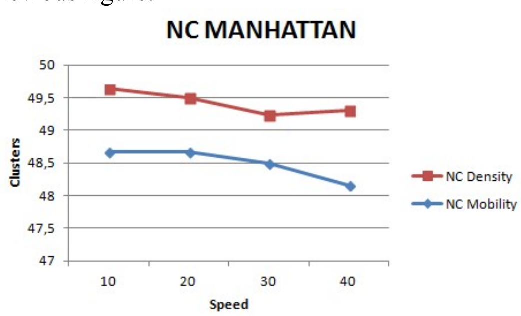

Figure 229: Average Number of Cluster = f (speed) In Manhattan

Figure 22 shows the evolution of the number of clusters in relation to the velocity of nodes in a network of 40 nodes using Manhattan as a mobility model.

We notice that the number of clusters in the algorithm based on mobility is less than the algorithm based on density.

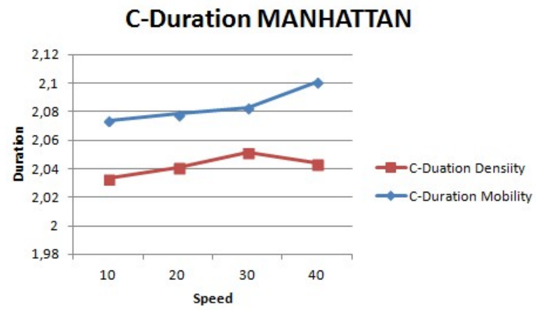

Figure 23: Average Cluster duration = f (speed) In Manhattan

Figure 23 shows the behavior of the average time during which a cluster is built based on the velocity of nodes in a network using Manhattan as a mobility model.

We notice that the mobility-based algorithm has the best duration results compared to the density based algorithm, and it's what was expected given the previous figure.

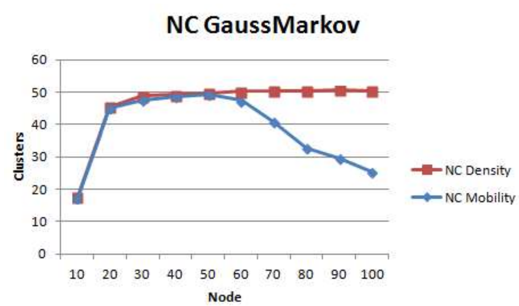

Figure 104: Average Number of Cluster = f (nbr nodes) In Gauss-Markov v=45m/s

Figure 24 shows the evolution of the number of clusters in relation to the number of nodes in a network using Gauss-Markov as a mobility model for a maximum speed of 45m /s.

We notice that the number of clusters in the algorithm based on mobility is less than the algorithm based on density especially when the number of nodes exceeds 50 nodes.

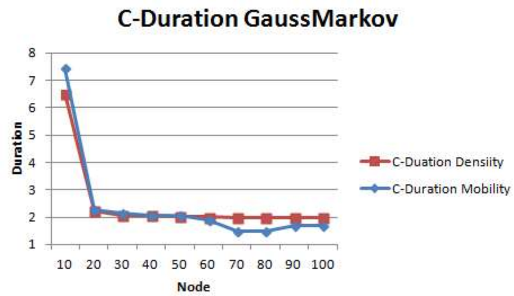

Figure 25: Average Cluster duration = f (nbr nodes) In Gauss-Markov V=45m/s

Figure 25 shows the behavior of the average time during which a cluster is built based on the number of nodes in a network using Gauss-Markov as a mobility model.

Contrary to what was expected, the duration of formation of cluster head becomes better in density-based algorithm once the node number exceeds 60 nodes despite the high number of clusters formed using this algorithm in the previous figure.

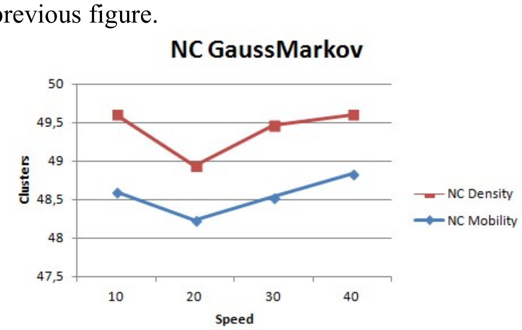

Figure 26: Average Number of Cluster = f (speed) In Gauss-Markov

Figure 26 shows the evolution of the number of clusters in relation to the velocity of nodes in a network of 40 nodes using Gauss-Markov as a mobility model.

We notice that the number of clusters in the algorithm based on mobility is less than the algorithm based on density.

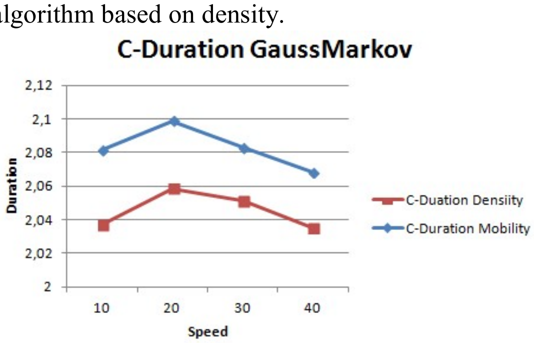

Figure 27: Average Cluster duration = f (speed) In Gauss-Markov

Figure 27 shows the behavior of the average time during which a cluster is built based on the velocity of nodes in a network using Gauss-Markov as a mobility model.

We notice that the mobility-based algorithm has the best duration results compared to the density based algorithm, and it's what was expected given the previous figure.

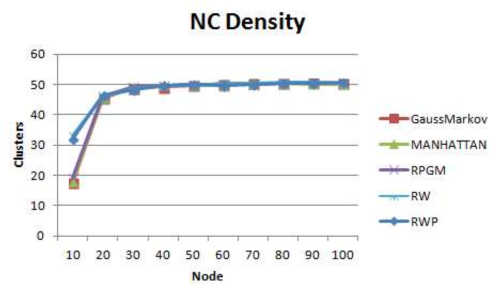

Figure 28: Average Number of Cluster = f (nbr nodes) using Density metric V= 45m/s

Figure 28 shows the number of clusters formed along the simulations in terms of the number of nodes in the network using our density-based algorithm.

We notice that our clustering solution gives approximately the same results for all the mobility models.

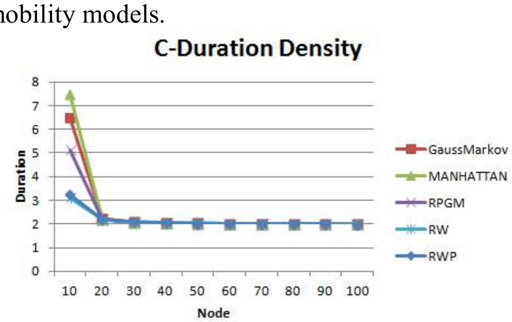

Figure 119: Average Cluster duration = f (nbr nodes) using Density metric V=45m/s

Figure 29 shows the duration of a cluster in terms of the number of nodes in the network using our Density-based algorithm.

We notice that the behavior of the clustering algorithm is practically the same for all models.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

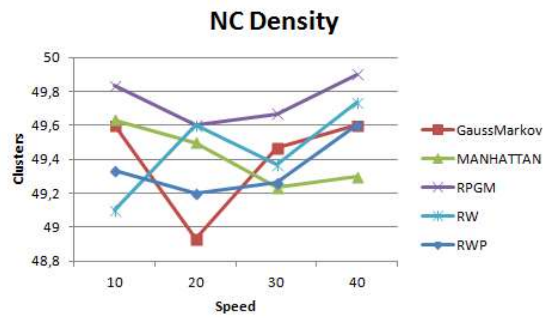

Figure 30: Average Number of Cluster = f (speed) using Density metric

Figure 30 shows the number of clusters formed during the simulation as a function of the maximum node speed for a network of 40 nodes using our Density-based algorithm.

We notice that Manhattan (The nodes move in this model in a much more organized way than in the other models) and RWP give the best results, the worst results were given by RPGM model

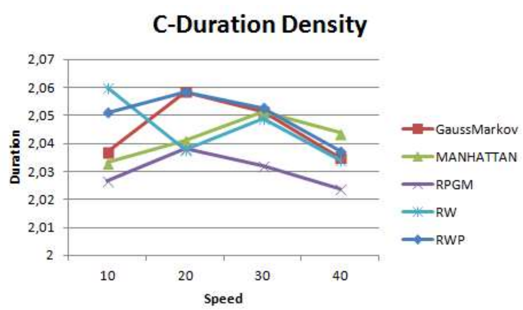

Figure 3112: Average Cluster duration = f (speed) using Density metric

Figure 31 shows the average lifetime of the clusters formed during the simulation as a function of the maximum node speed for a network of 40 nodes using our Density-based algorithm.

Still, and for the same reasons that we cited, the Manhattan and RWP models show the best results.

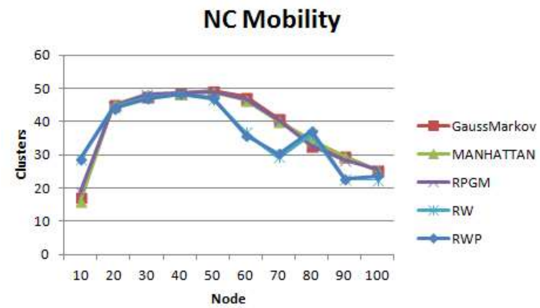

Figure 32: Average Number of Cluster = f (nbr nodes) using Mobility metric V= 45m/s

Figure 32 shows the number of clusters formed along the simulations in terms of the number of nodes in the network using our mobility-based algorithm.

We notice that our clustering solution gives approximately the same results for all the mobility models.

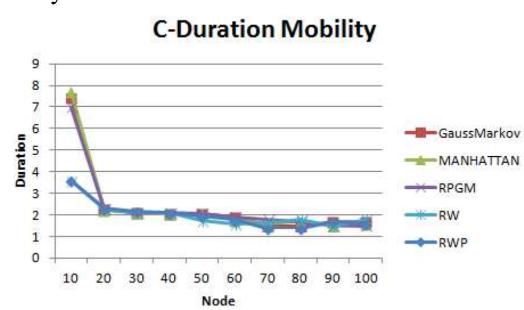

Figure 3313: Average Cluster duration = f (nbr nodes) using Mobility metric V=45m/s

Figure 33 shows the duration of a cluster in terms of the number of nodes in the network using our Mobility-based algorithm.

We notice that the behavior of the clustering algorithm is practically the same for all models.

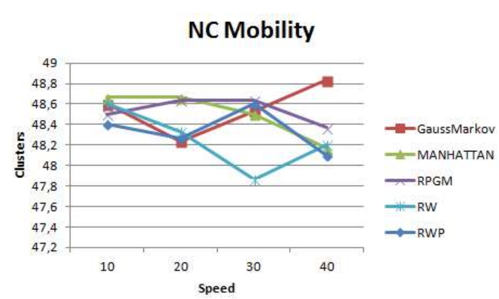

Figure 144: Average Number of Cluster = f (speed) using Mobility metric

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 34 shows the number of clusters formed during the simulation as a function of the maximum node speed for a network of 40 nodes using our Mobility-based algorithm.

We notice that RW gives the best results as it depend more on mobility compared to other models.

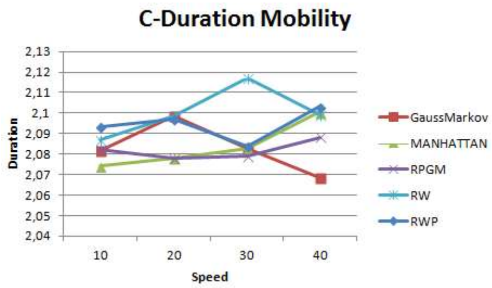

Figure 155: Average Cluster duration = f(speed) using Mobility metric

Figure 35 shows the average lifetime of the clusters formed during the simulation as a function of the maximum node speed for a network of 40 nodes using our Mobility-based algorithm.

Still, and for the same reasons that we cited, RW model shows the best results.

11. CONCLUSION:

Clustering is an important research topic for (MANETs) because clustering makes it possible to guarantee basic levels of system performance.

A large variety of approaches for ad hoc clustering has been presented.

In this work, we introduce an algorithm for efficient clustering of mobile ad-hoc networks.

Its contributions, compared to existing solutions, are summarized in the following: it does not add any new control message and the network is not overloaded or slowed at all, No changes are made to standard control messages. It works transparently with the AODV standard protocol.

Clusters are formed around the less mobile node; in other words, the node that has the lower mobility value is elected as cluster head for the mobility-based algorithm

In Density-based algorithm, Clusters are formed around the densest node; in other words, the node that has the largest number of neighbors is elected as cluster head

To make our algorithm more stable, we added the concept of the clustering interval, which represents the time at which each a cluster head, can be re-elected.

According to the results of simulations that we made, we notice a great improvement and better system stability with the adopted solution.

REFERENCES

[1] "Mobile Ad hoc Networking(MANET): Routing Protocol Performance Issues and Evaluation Considerations "Request for Comments 2501, IETF, January, 1999
[2] J. Jubin and J. D. Tornow, "The DARPA Packet Radio Network Protocols," Proceedings of the IEEE, Vol. 75, No. 1, pp. 21-32, Jan. 1987.
[3] B. M. Leiner, D. L. Neilson, F. A. Tobagui "Issues in Packet Radio Network Design," Proceedings of the IEEE, Vol. 75, No. 1, pp. 6-20, Jan. 1987.
[4] Sabina Baraković, Suad Kasapović, and Jasmina Baraković, "Comparison of MANET Routing Protocols in Different Traffic and Mobility Models", Telfor Journal, Vol. 2, No. 1, 2010.
[5] C. Perkins, B.-R. E. and D. S."Ad hoc On-demand Distance Vector routing" Request For Comments (Proposed Standard) 3561, Internet Engineering Task Force, July, 2003.
[6] http://datatracker.ietf.org/wg/manet/charter/
[7] C. Cordeiro, H. Gossain and D. Agrawal "Multicast over Wireless Mobile Ad Hoc Networks: Present and Future Directions" vol. 17, no. 1, pp. 52-59, January/February, 2003
[8] Sung-Ju Lee, Elizabeth M. Royer and Charles E. Perkins "Scalability Study of the Ad Hoc On-Demand Distance Vector Routing Protocol" In ACM/Wiley International Journal of Network Management, vol. 13, no. 2, pp. 97-114, March, 2003
[9] Ian Chakeres Elizabeth and M. Belding-Royer "AODV Implementation Design and Performance Evaluation" to appear in a

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

special issue on Wireless Ad Hoc Networkirng' of the InternationalJournal of Wireless and Mobile Computing (IJWMC), 2005

[10] David B. Johnson, David A. Maltz and Josh Broch "DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks" Proceedings of INMC, 2004 - cseweb.ucsd.edu

[11] Guoyou He. "Destination-sequenced distance vector (DSDV) protocol." Technical report, Helsinki University of Technology, Finland. 2 Dec 2003

[12] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark, "Scenario-based performance Analysis of Routing Protocols for Mobile Ad Hoc Networks," F'roc. ACM Mobicom 1999, Seattle WA, August 1999.

[13] K. Tamizarasu, M. Rajaram, "An AODV-based Clustering Approach for Efficient Routing in MANET", International Journal of Computer Applications (0975 – 8887) Volume 51– No.15, August 2012.

[14] Prithwish Basu, Naved Khan Thomas and D. C. Little "A mobility based metric for clustering in mobile ad hoc networks" Distributed Computing Systems Workshop, 2001

[15] M. SAADOUNE, A. HAJAMI and H. ALLALI "Local Distance's Neighbouring Quantification", International Journal of Wireless & Mobile Networks (IJWMN) Vol. 6, No. 6, December 2014

[16] M. SAADOUNE, A. HAJAMI and H. ALLALI "Relative speed quantification algorithm in AODV protocol", Global Summit on Computer & Information Technology 14-16 June 2014

[17] "The network simulator - ns-2"January, 2006

[18] T. Camp, J. Boleng, and V. Davies, "A Survey of Mobility Models for Ad Hoc Network Research", Wireless Communications & Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications, Vol. 2, No. 5. (2002), pp. 483- 502, 2002.

[19] Fan Bai, Narayanan Sadagopan, Ahmed Helmy. IMPORTANT: A framework to systematically analyze the Impact of Mobility on Performance of RouTing protocols for Adhoc NeTworks. Infocom‟03, April 1-3, 2003, San Francisco, California, USA.

[20] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang. A roup Mobility Model for Ad hoc Wireless Networks. In Proceedings of the ACM/IEEE MSWIM‟99, pp.53-60, Seattle, WA, August 1999.

[21] [94]. Geetha Jayakumar, Gopinath Ganapathi, " Reference point group mobility and random waypoint models in performance evaluation of MANET routing protocols " , Journal of Computer Systems, Networks, and Communications Volume 2008 , (January 2008)

[22] C. Park, Y. Lee, H. Yoon, S. Jin and D. Chio, "Cluster based Trust Evaluation in Ad Hoc Networks", pp. 503-507.

[23] L. Xu, X. Wang and J. Shen, "Strategy and Simulation of Trust Cluster Based Key Management Protocol for Ad hoc Networks", Proceedings of 4th International Conference on Computer Science & Education, (2009), pp. 269-274.

[24] S. Peng, W. Jia and G. Wang, "Voting-Based Clustering Algorithm with Subjective Trust and Stability in Mobile Ad-Hoc Networks", IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, (2008), pp. 3-9.