Classification Algorithm Based on Nodes Similarity for MANETs

Ali Choukri
Caddy Ayyad University, ENSA Safi, Morocco

Younes Hamzaoui(✉)
Ibn Tofail University, Kénitra, Morocco
Younes_ieee@hotmail.com

Mohammed Amnai
Sultan Moulay Slimane University, Beni-Mellal, Morocco
Ibn Tofail University, Kénitra, Morocco

Youssef Fakhri
Ibn Tofail University, Kénitra, Morocco

Abstract—This article describes an algorithm of classification by similarity of nodes in a MANET (Clustering). To optimize a network performance without influencing others, we must act only on the cluster structure. Any additional calculation clutters more the system. To overcome this limitation, a strong classification method is needed. The purpose of classification algorithms is the search for an optimal partition. This optimum is obtained iteratively refining an initial pattern randomly selected by reallocating objects around mobile centers. In order to partition the nodes into clusters, we used this technique (iterative reallocation) from the well known k-means algorithm. The algorithm conception is based on the k-means method that we improved and adapted to make it suitable for mobile ad hoc networks. The algorithm is implemented on OLSR giving birth to a new routing protocol: OLSRKmeans.

Keywords—MANETs; Classification; Routing; Clustering

1 Introduction

A mobile ad hoc network (MANET) is a special case of wireless networks where each node can directly join its neighbors using its radio interface. A node also has the ability to contact any others through intermediate nodes. These are responsible to relay messages and thereby provide a standalone network designed and supported by all participants.

Routing is a very important functionality of MANETs. The concept of routing is a set of procedures to ensure the establishment and maintenance of communication between the nodes.

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 clustering approach (classification). In this approach the close nodes are geographically grouped in a cluster. Moreover, this strategy uses different routing schemes in the same cluster (intra-cluster) and between clusters (inter-cluster). This technique 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.

2 Main Classification Methods

In this section, we present classification methods commonly used. These methods can be generally classified into two main families: by partitioning and by hierarchy.

2.1 Mobile centers method

The classification is an essential task for data analysis and interpretation. It is used to divide a set of data into groups or clusters. An item is characterized by a high similarity to components of its group, and a high dissimilarity to members of other groups [1]. There are two types of classification: supervised and unsupervised.

The supervised classification consists in classifying a new element in a class among a predefined set of classes starting from training data (pairs (element, cluster)). Membership of the n elements in different clusters is known. The goal is to build a classification rule to predict the cluster that contains a new element.

The unsupervised classification consists in dividing a heterogeneous set of data into sub-groups so that similar data are associated within same homogeneous group.

In the absence of any assumption about the distribution of nodes in a mobile ad hoc network, an unsupervised classification of nodes into clusters is required. We propose a model based on geometric considerations grouping: the geographically close nodes are grouped in a same cluster. This proposal requires a measure of closeness between nodes (distance). AISS journal's e-version (Volume 7, Number 2) is published on April 30th, 2015.

The most popular method of mobile centers is undoubtedly the k-means method [2] which is designed to process massive data sets. This method is based on simple algorithms that optimize the concept of inertia.

The k-means algorithm uses intra-class inertia based on a criterion called distance for classifying a set of objects described by variables. To classify a set of objects, it is common to use k-means with the X 2 metric [3].

2.2   Fuzzy clustering

Fuzzy c-means (FCM) is the fuzzy clustering methods most commonly used. It is an unsupervised classification method derived from c-means algorithm [4]. FCM is based on minimizing the sum of squares of the Euclidean distances between objects and classes centers.

2.3   Hierarchical classification

We mainly distinguish two types of families of hierarchical methods: ascending and descending. Descending method halve the class containing all objects. This operation is repeated for each iteration until all classes are reduced to singletons.

However, the ascending method agglomerates the closest classes. The end result is a single class containing all objects.

Unlike classification methods by partitioning (k-means and FCM), the use of hierarchical methods do not require knowledge of classes number.

3   Related Work

In this section, we present a few rare initiatives that have tried to exploit classification methods in mobile ad hoc networks.

The authors of [5] exploited the k-means method to calculate Meta-clusters by adapting it to a peer-to-peer environment. Each cluster is constructed as a weighted combination of source data. However, they had difficulty to specify beforehand the number of clusters.

In [6] the authors proposed an approach that combines the content-based subscriptions in a compact data format (Bloom lters) and ODMRP (On-Demand Multicast Routing Protocol). In order to build a mesh of dynamic scattering optimized, the approach is extended to use the context of the middleware layer. Dynamic multicast groups are created by aggregating subscriptions. The grouping method used by authors is k-means.

In Article [7] the authors based on the classification method k-means for study the strategy of a hidden Markov model (HMM) for intrusion detection using a multivariate Gaussian model for observations that are then used to predict an attack that exists in the form of a hidden state. This model consists of a network self-organized for event classification, a classifier observation, a drift detector, a profile estimator, an accelerator GMM (Gaussian Mixture Model) and an HMM engine.

The k-means method is also used by the authors of the article [8] to limit the flood search to a circular area around the local source in a MANET deploying RDMAR (Relative Distance Micro-discovery Ad Hoc Routing) by grouping nodes with similar characters in a same cluster.

Remi Badonnel and al. [9] have proposed an approach for monitoring errors in ad hoc networks. This approach is based on a measure of information theory adapted to ad hoc nodes intermittent and can detect network failures by inference. They defined a

distributed monitoring system with multiple detection methods of collaboration and a self-configuration mechanism based on k-means algorithm.

4 Algorithm Description

This work falls within the scope of research to conceive a strong MANET clustering algorithm that generates equitable and stable clusters. Thus this work represents an improvement of our previous contributions [10-12].

In the absence of any assumption about the distribution of nodes in a mobile ad hoc network, an unsupervised classification of nodes into classes is required. We propose a model based on geometric considerations (grouping geographically close nodes). This proposal requests define a measure of the proximity between nodes.

We begin by introducing the concepts of similarity, dissimilarity and distance between nodes, and then we will use the inertia concepts intra-class and inter-class to demonstrate mathematically how a classification method built homogeneous and distinct groups by translating it into a simple minimization problem.

C is a partition of X if and only if C satisfies the following properties:

- $C_a \subset$ X for all $C_a \in$ X
- $\bigcup_{a=1}^{t} C_a =$ C
- $C_a \cap C_b = \emptyset$ for (a, b) such that $a \neq b$

The property (3) expresses that the clusters formed are disjoint; each object of X belongs only to a single cluster of C .

To define the homogeneity of observations set, it is necessary to measure a similarity between two observations. Then we introduce the concepts of similarity and dissimilarity:

Dissimilarity is a function d which associates a value in IR + to each pair ($x_i x_j$) such that:

$$d(x_i, x_j) = d(x_j, x_i)$$

$$d(x_i, x_j) = 0 \rightarrow x_i = x_j$$

Conversely, another possibility is to measure the resemblance between observations using a similarity:

Similarity is a function s to which associates a value in IR* to each pair ($x_i x_j$) such that:

$$s(x_i, x_j) = s(x_j, x_i) \geq 0$$

$$s(x_i, x_j) \geq s(x_j, x_i)$$

The choice of the distance is a key issue for classification methods. To offer a relevant measure of similarity between elements, it is necessary to well use the available information at the nodes.

The Minkowski distance is the most used to determine the similarity between elements:

$$d(x_i, x_j) = \left(\sum_{k=1}^{P} \left| v_k(x_i) - v_k(x_j) \right|^l \right)^{1/l}$$

Where vk (xi) is the value of the object xi on the variable vk.
Depending on the values taken by the parameter l, we talk about:

- Euclidean distance (l = 2);
- Manhattan distance (l = 1);
- Chebychev distance (l = ∞).

We note that the metrics commonly used to analyze the ad hoc performances such as density, mobility and energy can be used to express distance.

Our objective is to divide the nodes into homogeneous and distinct clusters. To do this, we start from the definition of network inertia that can be modeled by a cloud of points.

Consider a network of n nodes ($x_1, \dots x_n$) and UG designates the centroid of nodes cloud

$$U_G = \frac{1}{n} \sum_{i=1}^{n} X_i \tag{1}$$

The cluster inertia is by definition the sum of squares of distances from the center:

$$I = \sum_{i=1}^{n} d^2 (x_i, u_G) \tag{2}$$

$$I = \sum_{i=1}^{n} \|x_i - u_G\|^2 \tag{3}$$

Assume that the network consists of P distinct clusters $C_1, , , C_P$. Each of these clusters having as centroid U_{ck} .

We can then decompose the total inertia of the cloud of nodes as follows:

$$I = \sum_{k=1}^{p} \sum_{i=1}^{n} \|x_i - u_G\|^2$$

$$I = \sum_{k=1}^{p} \sum_{i=1}^{n} \|x_i - u_{c_k} + u_{c_k} - u_G\|^2$$

(According to the theorem of Huygens)

$$I = \sum_{k=1}^{p} \sum_{i \in c_k} \left(\|x_i - u_{c_k}\|^2 + \|u_{c_k} - u_G\|^2 \right)$$

$$I = \left(\sum_{k=1}^{p} \sum_{i \in C_k} d^2(x_i, u_{C_K})\right) + \left(\sum_{k=1}^{p} d^2(x_{C_k}, u_G)\right) \tag{4}$$

The first term called intra-cluster inertia calculates the sum of distances between nodes and their centroid. Low intra-cluster inertia indicates that the nodes in the same cluster are more nearer (Clusters are homogeneous).

The second term called the inter-cluster inertia calculates the sum of distances between centroids of clusters and global centroid, that is to say the separation degree of clusters.

$$I = I_{intra} + I_{inter} \tag{5}$$

From a formal point of view, the optimal partition is:

- That minimizes the intra class inertia
- Or that maximizes the inter class inertia,

Thus the optimal partition would be defined as follows:

$$C_k^{optimal} = Argimin_{C \in C_k} \sum_{k=1}^{p} \sum_{i \in C_k} d^2(x_i, u_G) \tag{6}$$

The objective of classification algorithms based on this principle is the search of optimal partition.

In practice it is impossible to generate all patterns of clustering for evident reasons of complexity. We then seek a scheme sufficiently close to the optimum. This optimum is obtained in an iterative manner by improving an initial scheme randomly selected by reallocating objects around mobile centers.

In order to partition the nodes into clusters, we used this technique (iterative reallocation) based on k-means algorithm

4.1 K-means method

The k-means algorithm [13] is a partitioning method widely used in various application areas.

From P separate clusters, the k-means algorithm assigns iteratively objects $(x_1, \dots x_n)$ at P centers of clusters ($u_1, \dots u_P$), followed by calculating the positions of the new centers . The stopping algorithm is a criterion fixed by the user and can be:

- Achieve a limited number of iterations;
- The algorithm converges: clusters formed remain the same between two successive iterations;

The inertia intra-cluster is not improving between two iterations (the algorithm is sufficiently close to convergence).

To justify the K-means algorithm in view of our goal aimed to minimize the intra-cluster inertia, we demonstrate that the redefinition of centers of clusters and the real-

location of nodes (Sequence of two sub-stages of K-means algorithm) results in a decrease in intra-class inertia.

We start from following considerations at the end of determined step j:

- Centers of clusters $\left(u_1^j, \dots u_p^j\right)$ have been calculated;
- The classes $\left(C_1^j, \dots C_p^j\right)$ were obtained by assigning at the center $\left(C_k^j\right)$ the n_j^i nearest nodes.

We define the following quality at the end of step j:

$$Q_j = \sum_{k=1} \sum_{i \in C_k^j} \left\| x_i - u_k^j \right\|^2$$

The redefinition sub-step of new centers and reassigning of nodes (next iteration) requires:

Recalculate centers of clusters $\left(u_1^{j+1}, \dots u_p^{j+1}\right)$ based on points belong to each of clusters $\left(C_1^{j+1}, \dots C_p^{j+1}\right)$ possessing respectively n_k^j element. We have:

$$u_k^{j+1} = \frac{1}{n_k^j} \sum_{i \in C_k^j} x_i$$

And:

$$W_{j+1} = \sum_{k=1}^{p} \sum_{i \in C_k^j} \left\| x_i - u_k^{j+1} \right\|^2$$

Where u_k^{j+1} is the center of the cluster C_k^j et W_{j+1} is the inetria intra-clusters clusters assosiated to clusters $\left(C_1^j, \dots C_p^j\right)$. We will have:

$$Q_i = \sum_{k=1}^{p} \sum_{i \in C_k^j} \left\| x_i - u_k^j \right\|^2$$

$$Q_i = \sum_{k=1}^{p} \sum_{i \in C_k^j} \left\| x_i - u_k^{j+1} + u_k^{j+1} - u_k^j \right\|^2$$

$$I = \sum_{k=1}^{p} \sum_{i \in C_k^j} \left(\left\| x_i - u_k^{j+1} \right\|^2 + \left\| u_k^{j+1} - u_k^j \right\|^2 + 2 \langle x_i - u_k^{j+1}, u_k^{j+1} - u_k^j \rangle \right)$$

$$I = W_{j+1} + \sum_{k=1}^{p} \sum_{i \in C_k^j} \left\| u_k^{j+1} - u_k^j \right\|^2 + 2 \sum_{k=1}^{p} \langle \sum_{i \in C_k^j} (x_i - u_k^{j+1}), u_k^{j+1} - u_k^i \rangle \geq W_{j+1}$$

The vector $\sum_{i \in C_k}(x_i - u_k^{j+1})$ being the null vector by definition of u_k^{j+1}

Reassign nodes to the nearest centers. Then we obtain new clusters $\left(C_1^{j+1}, \dots C_p^{j+1}\right)$ and we defin:

$$Q_{j+1} = \sum_{k=1}^{p} \sum_{i \in C_k^j} \left\| x_i - u_k^{j+1} \right\|^2$$

After reassignment all distances decrease because each node xi is assigned to the cluster center minimizing thereby the gap: $\left\| x_i - u_k^{j+1} \right\|^2$

We therefore have:

$$Q_{j+1} \leq W_{j+1}$$

Thus, for each j we have proved the following inequality:

$$Q_{j+1} \leq W_{j+1} \leq Q_j$$

So we have in particular:

$$W_{j+2} \leq Q_{j+1} \leq W_{j+1}$$

This shows that after each algorithm iteration the improvement of the nodes classification is sure within the meaning of I_{intra} criterion.

Because the intra-cluster inertia of the optimal partition is the smaller, the margin of improvement is finite. This implies that the algorithm converges inevitably.

The disadvantage of the k-means method is that the number of clusters is a parameter of the algorithm. This is not obvious for a mobile ad hoc network where nodes can join or leave the network randomly. Hence the need to make improvements to the method to be applicable to our case:

- **Initial number of Clusters**: At the beginning each node represents its own cluster. Thereafter we make a series of ascending partitions by combining the nodes belonging to the same neighborhood into same cluster until reaching a stable number of clusters.
- **Distance:** In a network where nodes are characterized by similar radio ranges (The case of most simulations in NS2), the node with the highest density is the center of its neighborhood. So we will use the density of nodes to express the distance metric.

We present below our algorithm in detail (k-means improved to make it applicable to the partition of nodes into clusters):

Proposed algorithm

Input: Mobile ad hoc network of n nodes.

Output: Network virtually partitioned into P clusters

Distance (d): The density of nodes is the dissimilarity measure we have used.

Step 0

1. Initialization with n centers $(u_1^0 \ldots u_n^0)$ each node is a cluster

2. Creation of an initial partition P0 $=\{(C_1^0 \ldots C_k^0)\}$

Initialize l to 1 (l is iteration index)

Assign to u_l^0 its is two-hop neigbours ;

$C_l^0 = \{x_i \in$ network $\mid d(x_i, u_l^0) >= 2hops\}$;

Remove from list of centers the $n_{C_l^0}$ nodes assigned to centroid u_l^0;

Move to the medoid $l + n_{C_l^0}$

Repeating steps b to d until all node are affected ;

Calculation of new centers of k cluster obtained $(u_1^1 \ldots u_k^1)$: nodes having greatest density within their clusters.

Step t

3. creation of new partition Pt $=\{(C_1^t \ldots C_k^t)\}$ by assigning to each centers its two-hop neighbors;

4. the centers affected to other centers are removed from the list of centers;

5. the isolated nodes are assigned to the list of centers ;

6. Calculate the centers of k clusters obtained $(u_1^0 \ldots u_k^0)$;

7. Repeat steps 3 to 6 until that a stable partition is achieved (structure of partition P_{t+i} equals that the P_{t+i+1}) or reach n iterations.

The experiments that we conducted to test the complexity of the algorithm have shown that the number of iterations required to converge is low (generally less than 10). This indicates that the algorithm is adapted to the classification of large-scale networks (several thousands of nodes).

5   Simulations Results

The simulations are performed in the networks simulator NS2: The mobility model used is Random Waypoint with a maximum speed of 10 m/s. The density of the network is varied from 10 to 100 nodes.

The simulation parameters used in each layer are shown below:

- **Application layer**: The video support is the sequence "Foreman" in QCIF format (176 ×144 pixels/picture). A sequence of 300 pictures of the video is encoded at 15 pictures/s by using the H.264 video encoder profile.
- **Routing layer**: routing in the network is provided by the protocols OLSRMax-Min2C [10] and OLSRkmeans.
- **Physical layer**: The transmission parameters (power and transmission frequency, sensitivity threshold) were determined according to 802.11.

Our new algorithm is implemented on the OLSR routing protocol. Because of this we named it OLSRkmeans.

OLSRkmeans is evaluated by comparing it to our previous contribution OLSRMaxMin2C [10]. The simulations were performed based multiservice traffic (H.264) in real time.

Performance comparison of the two algorithms will be started by the analysis of their stabilities by examining the number of generated clusters during the simulation period. Thereafter we compare it in terms of end to end delay, packet delivery ratio and routing control overhead.

5.1 Number of generated clusters

A minimum number of clusters means that each cluster contains a maximum of nodes and each node is eccentric in its neighborhood (surrounded by a reasonable number of neighbors).
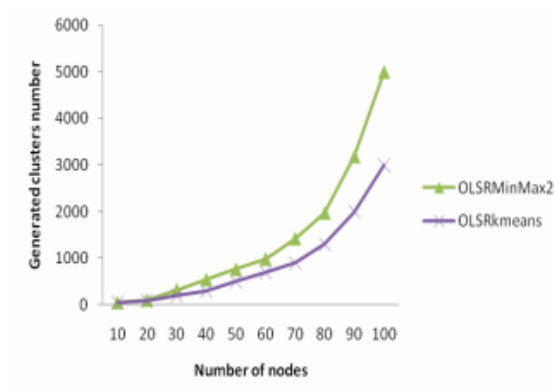
Fig. 1. Number of generated clusters according to number of nodes

Figure 1 shows the evolution of number of generated clusters by OLSRMaxMin2C and by OLSRkmeans according to the density of network whose nodes are characterized by a maximum speed of 10 m/s and a simulation area of 500*500 m.

We note that OLSRkmeans generates fewer clusters compared to OLSRMaxMin2C especially for denser networks.

This means that the clusters generated by OLSRkmeans are denser in comparison to those generated by OLSRMaxMin2C. This leads to conclude that the clusters structure formed by OLSRkmeans is more stable.

This result can be interpreted by the structure of kmeans that assigns nodes to the nearest centers and recalculates the new centers alternately until it reaches a partition very close to the optimal.

5.2   End to end delay

The estimate of end to end delay is essential in view of designing a routing protocol that respects the constraints of multimedia applications.

The following figure shows the evaluation of protocols OLSRkmeans and OLSRMaxMin2C in terms of the average duration needed to route a packet from a source to a destination.

We recall that the average end to end delay is equal to sum of the delays on the links that make up a path.

This time takes account two main steps: the delay in tail of nodes along the path and the packet propagation delay on the physical medium.
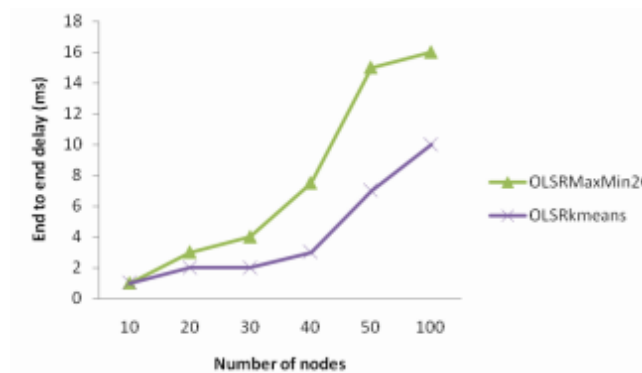

Fig. 2. Average end to end delay according to number of nodes

We observe that the time required to transmit a packet from the source to the destination with the OLSRkmeans protocol is reduced, especially for dense networks.

This can be interpreted by the fact that OLSRkmeans forms less clusters than OLSRMaxMin2C and consequently less clusterheads.

The fact that the inter-cluster interactions are limited to clusterheads helped reduce the size of the paths from source to destination in terms of number of hops and consequently to improve the average transmission delay.

5.3   Packet delivery ratio

In an ad hoc network, several events may cause packet loss such as congestion in the network, a high error rate or a broken link following the mobility of one or more nodes.

Packet losses, as mentioned above, influence performances of multimedia applications. Hence the need to evaluate a routing protocol in terms of packet delivery ratio.

The following figure shows the comparison of protocols OLSRkmeans and OLSRMaxMin2C based on a VBR stream (H.264) in terms of packet delivery ratio and according to network density.
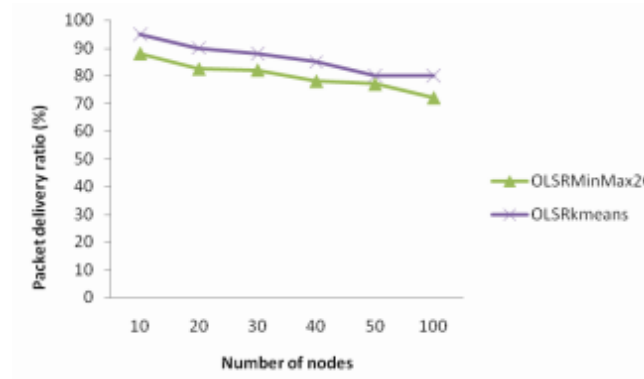
Fig. 3. Packet delivery ratio according to network density

We observe that OLSRkmeans reduces the number of packets lost compared to OLSRMaxMin2C.

This result implies that OLSRkmeans tends to improve throughput that is organically linked to the losses. In addition, we can conclude that it can also optimize energy consumption by saving that required for retransmission of lost packets

5.4 Control Routing Overhead

Users of ad hoc networks generally use applications transmitting VBR traffic (video and sound) in real time. To support the QoS required by these applications, it is necessary to take into consideration the traffic load on the network to speed delivery of information.

Meet the constraints on these applications is more difficult in an ad hoc network due to the high variability of bandwidth. It is therefore necessary to minimize as much as possible the amount of traffic control in favor of real data.
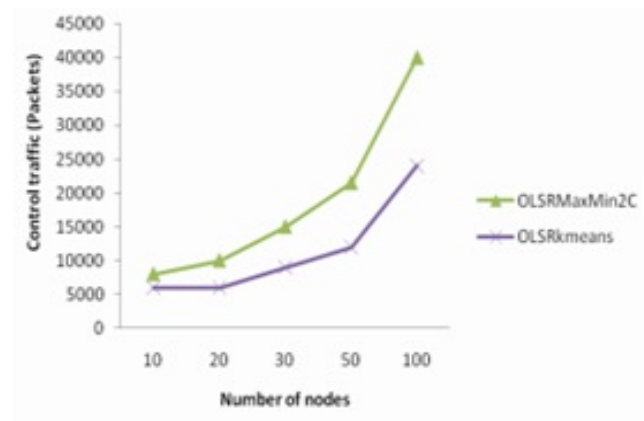

Fig. 4. Control traffic overhead according to network density

Figure 4 shows that OLSRkmeans reduced considerably the number of control packets exchanged between nodes relative to OLSRMinMax2C especially for large scale networks. This constitutes a gain of bandwidth.

This should certainly positively influence other performances such as throughput and PDR (packet delivery ratio).

This result can be interpreted by density of clusters generated by OLSRkmeans which exceeds that of clusters formed by OLSRMaxMin2C.

6 Conclusion

In this article, we presented the classification algorithm k-means based on minimization of differences between elements of a cluster and maximization of distance between clusters (intra-cluster and inter-cluster Inertia).

We confirmed that k-means improves the classification of nodes by demonstrating theoretically that the method reduces the intra-cluster inertia (therefore it increases the inter-cluster inertia) between two successive iterations.

We also showed that an optimal partition exists (at least theoretically), and the algorithm converges inevitably.

Subsequently, we implemented the method on OLSR giving birth to a new routing protocol called OLSRkmeans.

In order to evaluate OLSRkmeans we compared it to our previous contribution OLSRMaxMin2C. Simulations have shown that the results obtained by OLSRkmeans exceed those by OLSRMinMax2C in terms of stability (number of nodes per cluster), control traffic, delay and packet delivery ratio.

7 References

[1] Daniel T. Larose, "Discovering Knowledge in Data: An Introduction to Data Mining", John Wiley & Sons, Inc., Hoboken, New Jersey. 2005

[2] Forgy, E. W., "Cluster Analysis of Multivariate Data: Eficiency versus Interpretability Models". Biometrics, 61(3):768_769, 1965.

[3] Benzecri, J. P., "L'analyse des données : la taxinomie", Tome 1. Dunod, Paris, 1973.

[4] Ball, G. H. and Hall, D. J. ISODATA, "an Iterative Method of Multivariate Analysis and Pattern Recognition". Behavior Science, 153, 1967.https://doi.org/10.1002/bs.3830120210

[5] Sandor Dornbush and Anupam Joshi, StreetSmart Traffic: Discovering and Disseminating Automobile Congestion Using VANET's, Vehicular Technology Conference, 2007. VTC2007 Spring. IEEE 65th, pp 11 – 15, 2007. [6] Eiko Yoneki and Jean Bacon, Distributed Multicast Grouping for Publish/Subscribe over Mobile Ad Hoc Networks, Wireless Communications and Networking Conference, 2005 IEEE, pp 2293 - 2299 Vol. 4, 2005.

[6] Rahul Khanna and Huaping Liu, System Approach to Intrusion Detection Using Hidden Markov Model, IWCMC '06 Proceedings of the 2006 international conference on Wireless communications and mobile computing, pp 349-354, ACM 2006.

[7] Hamad, O.F, .Mi-Young Kang ; Jin-Han Jeon ; Ji-Seung Nam, Neural Network's k-means Distance-Based Nodes-Clustering for Enhanced RDMAR Protocol in a MANET, Signal Processing and Information Technology, IEEE 2008, pp 192 – 197, 2008.

[8] Remi Badonnel, Radu State, Olivier Festor, Self-configurable fault monitoring in ad-hoc networks, Ad Hoc Networks Volume 6, Issue 3, pp 458–473, May 2008. https://doi.org/10.1016/j.adhoc.2007.04.001

[9] Ali CHOUKRI, Ahmed HABBANI, Mohammed EL KOUTBI, "A hierarchical version of OLSR for MANET", World Applied Sciences Journal 21 (12) , pp. 1739 2013.

[10] Ali CHOUKRI, Ahmed HABBANI, Mohammed EL KOUTBI, "Efficient heuristic based on clustering approach for OLSR", Journal of Computer Networks and Communications 2013.

[11] Ali CHOUKRI, Ahmed HABBANI, Mohammed EL KOUTBI, "An Energy Efficient Clustering Algorithm for MANETs", ICMCS 2014.

[12] MacQueen J., "Some methods for classification and analysis of multivariate observations". In: Proceedings of the Fifth Berkeley Symposium on Mathematical statistics and probability. pp. 281-297. Berkeley, 1967.

8 Authors

Ali Choukri is an Assistant professor at National School of Applied Sciences. He was awarded a Master of Computer Science and Telecom from the University of Ibn Tofail, Kénitra-Morocco in 2008. He got ENSET (higher Normal School of technical teaching) degree in 1992. He has a PhD from the School of Computer Science and Systems Analysis (ENSIAS). He is working within the MIS team of the Laboratory SIME, for studying ad hoc mobile intelligent communication systems, and wireless sensor networks. His research interests are in the areas of ubiquitous computing, Internet of Things, Delay/Disruption Tolerant Networks, Wireless Networks, QoS routing, Mathematical modeling and performance analysis of networks, Control and decision theory, Game theory, trust and reputation management, Distributed algorithms, Metaheurtistics and optimization, Genetic algorithms.

Younes Hamzaoui obtained his DEUG degree on phsysics chemical in 2000 from Abdmalik saadi University, Tanger city, morocco. Then, he received his degree on IEEA (Computers, Electronics, Electrical and Automation) in 2003. After in 2007, he obtained his DESA degree on information system and telecommunication. The authers is a professor of computer science in high school. He is a PhD student of management of Qos in routing protocol, also, he doing his research in the Laboratory in Computer Science and Telecomunications (LaRIT).

Mohamed Amnai received his bachelor's degree in 2000, in IEEA (Computers, Electronics, Electrical and Automation) from Molay Ismail's University, Errachidia city. Then, he obtained his master's degree in 2007, from Ibn Tofail University, Kenitra city. In 2011, he received his PhD on Telecommunication and computer science, from Ibn Tofail University in Kenitra city, Morocco. Since March 2014 he is a Professor, he has been an Assistant at National School of Applied Sciences Khouribga, Settat University, Morocco. The author is also an associate member of Research Laboratory in Computer Science and Telecommunications (LARIT), Team Networks and Telecommunications Faculty of Science, Kenitra, Morocco. Also, the author is an associate member of laboratory IPOSI National School of Applied Sciences (ENSA) Khouribga, Sultan Moulay Slimane University, Morocco.

Youssef Fakhri received his Bachelor's Degree (B.S) in Electronic Physics in 2001 and his Master's Degree (DESA) in Computer and Telecommunication from the Faculty of Sciences, University Mohammed V, Rabat, Morocco, in 2003 where he developed his Master's Project at the ICI Company, Morocco. He received a PhD in 2007 from the University Mohammed V - Agdal, Rabat, Morocco in collaboration with the Polytechnic University of Catalonia (UPC), Spain. He joined the Faculty of Sciences of Kénitra, Department of Computer Science and Mathematics, IbnTofail University, Morocco, as an Associate Professor in March 2009, he is the Laboratory head at LaRIT (Laboratory for Research in Computing and Telecommunications) in the Faculty of Kénitra, and Member of Pole of Competences STIC Morocco.