An Efficient Scalable Weighted Clustering Algorithm for Mobile Ad Hoc Networks

Mohamed Aissa[1], Abdelfettah Belghith[2]
[1]University of Nizwa
Nizwa Sultanate of Oman
[2]Hana Research Group
ENSI University of Manouba - Tunisia
m.issa@unizwa.edu.om, abdelfattah.belghith@ensi.rnu.tn

ABSTRACT: *We consider the problem of appropriate clusterhead selection in wireless ad-hoc networks where it is necessary to provide robustness in the face of topological changes caused by node motion, node failure and node insertion/removal. The main contribution of our work is a new strategy for clustering a wireless AD HOC network and improvements in WCA. We first derived a simple stability model and thereafter a load balancing clustering scheme. We showed that our algorithm outperforms the Weighted Clustering Algorithm (WCA) in terms of cluster formation and stability. One of the main ideas of our approach is to avoid clusterhead re-election and to reduce the computation and communication costs by implementing a non-periodic procedure for clusterhead election which is invoked on-demand. We strived to provide a trade-off between the uniformity of the load handled by the clusterheads and the connectivity of the network.*

Keywords: Ad hoc network, Clusters, Load-balacing, Stability

Received: 23 November 2012, Revised 4 January 2013, Accepted 8 January 2013

1. Introduction

Ad hoc networks are wireless, infrastructureless, multi-hop, dynamic networks established by a collection of mobile nodes, which provides significant features to the modern communication technologies and services [1]. In ad hoc networks, clustering is an important and familiar technique to divide the large network into several sub networks.

Cluster-based routing is a solution to address node heterogeneity, and to limit the amount of routing information that propagates inside the network [2, 17, 18, 19]. The idea behind clustering is to group the network nodes into a number of overlapping clusters. Clustering makes possible a hierarchical routing in which paths are recorded between clusters instead of between nodes. This increases the routes lifetime, thus decreasing the amount of routing control overhead. Inside the cluster, one node that coordinates the cluster activities is known as a clusterhead (CH). The set of clusterheads is known as a *dominant set*. Inside the cluster, there are also ordinary nodes and have direct access only to this one clusterhead, and gateways. Gateways are nodes that can hear two or more clusterheads [2]. Groups of the nodes are organized with respect to their nearness to other nodes. Two nodes are said to be neighbors of each other when both of them lie within their transmission range and set up a bidirectional link between them [3]. Clustering is an important approach to solving capacity and scalability problems in mobile ad hoc networks where no physical infrastructure is available. The connected dominating set (CDS) is a special cluster structure in which the cluster heads form a connected network without using gateways. A clusterhead does the resource allocation to all the nodes belonging to its cluster. Due to the dynamic nature of the mobile nodes, their association and dissociation to and from clusters disrupt the stability of the network,. and thus reconfiguration of clusterheads is unavoidable. This is an important issue since frequent clusterhead changes adversely affect the performance of other protocols such as scheduling, routing and resource allocation that rely on it. The choice of the clusterheads is here based on the weight associated to each node: the bigger the weight of a node, the better that node is for the role of clusterhead.

In [4], the authors have proposed a distributed weighted clustering algorithm by making some modifications and improvements on some existing algorithms. They demonstrated that their algorithm reduces the clusterhead formation and control messages overhead thus improving overall performance and reducing energy utilization. Here, authors claimed that since energy utilization is the most important criteria in cluster based routing schemes, their protocol provides better results than existing distributed clustering algorithms.

A weight based distributed clustering algorithm (WCA) which can dynamically adapt itself with the ever changing topology of ad hoc networks was proposed in [5]. In this approach, the number of nodes is restricted to be catered by a cluster head, so that it does not degrade the MAC functioning. It has also the flexibility of assigning different weights and takes into account a combined effect of the ideal degree, transmission power, mobility and battery power of the nodes. The algorithm is executed only when there is a demand, i.e., when a node is no longer able to attach itself to any of the existing cluster heads. Clustering algorithm tries to distribute the load as much as possible. In WCA, the number of nodes that a clusterhead can handle ideally is δ. This is to ensure that clusterheads are not over-loaded and the efficiency of the system is maintained at the expected level. But how to select δ is not addressed explicitly. If d is not well selected, many clusterheads are generated which increases energy consumption. Furthermore, computing the clusterhead serving time cannot guarantee a good assessment of energy consumption, because data communication consumes a large amount of energy and varies greatly from node to node. Our contribution is to extend WCA by solving these inefficiencies.

In [6], using a heuristic approach, the authors give some interesting equations for the cluster density and cluster order of homogeneously distributed nodes running the DMAC algorithm [7]. Since the DMAC structure is unique, the equations also hold in a mobile scenario if the mobility model used retains the homogeneous distribution of the nodes. If the nodes are not homogeneously distributed, the cluster density will decrease. The authors claim that the validity of their result is not restricted to the DMAC algorithm. It also holds for other algorithms that limit the cluster size to two hops.

In [8], the authors introduced a new type of algorithm called Enhancement on Weighted Clustering Algorithm [EWCA] to improve the load balancing, and the stability in the MANET. The cluster head selected efficiently based on these factors, like high transmission power, transmission range, distance mobility, battery power and energy. Since the cluster head will not be changed dynamically, the average number of cluster formations will be reduced. By applying the load balancing factor, the overhead in the cluster is reduced.

In [9], the authors develop a clustering algorithm usable by large scale ad hoc networks with high mobility. The main objective of their algorithm is ensure the stability of the clusters and to reduce the re-election of the cluster head. They introduce some metrics to choose the cluster head in order to respect the capacities of the node and to reflect the state of the network. They claim that these metrics merge together and provide a higher connectivity and economy of energy, as well as the best value of transmission range. The authors propose a mobility scheme to evaluate periodically the node mobility in order to expect the future state of the network. However, this scheme is complicated and requires overhead calculations.

The motivation for the present work is three-fold. First, we have identified some weakness in [5, 8, 10, 11, 12] where the authors declared that according to their notation, the number of nodes that a clusterhead can handle is ideally constrained by a value δ. Second, we have identified another weakness in [5, 8, 10, 11, 12], where the authors compute the *degree-difference* for every node to ensure that clusterheads are not over-loaded. Third**,** the stability is overlooked in WCA. Consequently, we introduce our analytical models to overcome the previous inefficiencies.

In the remainder of this paper, Section 2 presents the network model and problem specifications. Our algorithm analytical model is given in Section 3. The formal definition of the SWCA algorithm and its illustrative example are given in Section 4. Conclusions are given in Section 5.

Next, we formulate our network model as it was defined by Chatterjee, Das, and Turgut [5].

2. Network Model and Problem Specifications

As defined in [5], the network formed by the nodes and the links can be represented by an undirected graph $G = (V, E)$, where V represents the set of nodes v_i and E represents the set of links e_i . Note that the cardinality of V ($|V|$) remains the same but $|E|$ always changes with the creation and deletion of links.

Clustering can be thought as a graph partitioning problem with some added constraints. As the underlying graph does not show any regular structure, partitioning the graph optimally (i.e., with minimum number of partitions) with respect to certain parameters becomes an NP-hard problem [13]. The neighborhood $N(v_i)$ of a clusterhead v_i is the set of nodes which are directly linked to it and which are in fact the nodes lying within its transmission range. This defines the degree of the node v_i.

$$N(v_i) = \{v_j,\ such\ taht\ \ dist(v_i, v_j) < R_{v_i}\} \quad (1)$$

where $dist(v_i, v_j)$ is the measured average distance between v_i and v_j.

The node degree of a node v_i is deduced as the cardinality of the set $N(v_i)$:

$$deg(v_i) = |N(v_i)| \quad (2)$$

More formally, we are looking for the set of vertices $S \subseteq V(G)$, such that the union of $N(v_i)$ where $v_i \in S$ forms $V(G)$. The set S is called a *dominating set* such that every vertex of G belongs to S or has a neighbor in S.

In order to meet the requirements imposed by the wireless mobile nature of these networks, a clustering algorithm is required to partition the nodes of the network so that the following ad hoc clustering properties are satisfied [7]: (a) Every ordinary node has at least one clusterhead as neighbor (*dominance property*), (b) Every ordinary node affiliates with the neighboring clusterhead that has the smaller weight, and (c) No two clusterheads can be neighbors (*independence property*). Next, we propose our algorithm analytical model.

3. SWCA Model

In our proposed SWCA (*Scalable Weighted Clustered Algorithm*), we propose two new models in clustering algorithms: node stability and load balancing models.

3.1 Clustering stability enhancement

Despite the fact that node mobility is an intrinsic characteristic in MANETs, the cluster structure should be maintained as stable as possible [12]. Otherwise, frequent cluster change or re-clustering adversely affects the performance of radio resource allocation and scheduling protocols [12]. By stability, we mean that the cluster structure remains unchanged for a given reasonable time period [12]. Consequently, stability is an important requirement, which should be taken into consideration in our clustering algorithm.

In [5], the authors claimed that a node with less mobility is always a better choice for a clusterhead. For this purpose, they computed the running average of the speed for every node v_i till current time T, and deduced a measure of mobility based on the coordinates of the node v_i at current and previous times. This can lead to calculations overheads and inaccuracies due to mobility nodes. The majority of link stability models envisioned in the literature is based on node remoteness either directly or indirectly. To calculate the node stability of a node , we are motivated by the results conducted in [14]. Starting from the principle that stability increases with the remoteness between the two end vertices, the authors simply defined the stability of a link e with incident vertices i and j, as a linear function of the current distance (transmission range) between two nodes v_i and v_j (d_{ij}). Hence the stability $\psi(e)$ of an edge e verifies:

$$lim_{d_{ij}\to 0}\ \psi(e) = 1,\ lim_{d_{ij}\to\infty}\psi(e) = 0 \quad (3)$$

Starting from (3), we propose our new stability scheme based on the transmission zone aggregation depicted in Figure 1. The transmission range of a node v_i forms a circle with a radius r_2 (circle 1). Inside this circle, the neighbor nodes of can exist within an inner circle with a radius r_1 (circle 2). This circle forms a trusted zone where these neighbor nodes are considered as trusted neighbors whose neighborhood is guaranteed for a well-defined period. We call these nodes favorable nodes. However, other neighbor nodes can be situated between circle 1 and circle 2. We call this zone the *risked zone*. The width of this zone is $l = r_2 - r_1$ The neighbor nodes situated in this zone are considered as topologically unfavorable nodes because they can be assumed to leave the partition earlier than more centralized ones which are situated in the trusted zone. We call these nodes unfavorable nodes.

Our contribution is to translate Figure 1 into a mathematical scheme, which once incorporated in our proposed algorithm should give higher priority to favorable nodes and less priority to unfavorable nodes during clusterhead selection processes. For this purpose, we introduce *a range indicator* (*rind*) which is calculated as follows:

$$rind\,(v_i, v_j) = \begin{cases} 1, & if \;\; dist\,(v_i, v_j) \le r_1 \\ \alpha\,(r_2 - r_1), & if \; r_1 < dist\,(v_i, v_j) < r_2 \end{cases} \qquad (4)$$

where $0 < \alpha < 1$ is input user coefficient which can be tuned by choosing the suitable values based on the network mobility rate and *dist* (v_i, v_j) the measured average distance between the nodes v_i and v_j.

From (4), we can deduce that if $r_1 = r_2$, then *rind* $(v_i, v_j) = 0$ Actually, this range indicator can help in identifying topologically favorable and unfavorable nodes.

So far, we have been dealing with measured average distances. However, to benefit from our proposed range indicator, we introduce a new measure which is "*virtual range distance*" from a node v_i to a node v_j and calculated as follows:

$$virdist\,(v_i, v_j) = rind\,(v_i, v_j) \times dist\,(v_i, v_j) \qquad (5)$$

Consequently, we are motivated to calculate the virtual distance ($D\,(v_i)$) from one node v_i to all the set of its neighbors *which* are direct linked to it (situated within its transmission range (R_{v_i})).

$$virD\,(v_i) = \sum_{j=1}^{n} virdist\,(v_i, v_j) \qquad (6)$$

Based on the previous equations, we set our stability factor for each node v_i as:

$$STF\,(v_i) = vir\,D\,(v_i) \,/\, deg\,(v_i) \qquad (7)$$

In our proposed SWCA algorithm, the neighbor nodes with higher stability factor *stf* (v_i) are considered good candidates to be selected as clusterheads.

3.1.1 Observation

It is very interesting to mention that (7) dictates that a clusterhead should be situated as a core node in a secure zone.

It is worth mentioning that the stability of the clustered topology can be achieved by reducing significantly on the number of clusters formed and the number of re-affiliations under different scenarios. Our next problem consists of proposing our new cluster degree constraint and our modified degree-difference schemes to replace those claimed in [5, 8, 10, 11, 12].

3.2 Load balancing clustering scheme

A system can have patches of high-density clusters and very low-density clusters [15].

In such scenarios the high-density clusterhead will be overwhelmed with processing and communication load, and will consume its energy quickly, while the low density cluster-head will sit idle wasting precious time [15]. A higher than ideal degree causes latency in information delivery [16]. At the same time, it is difficult to maintain a perfect load balanced system at all times, due to frequent detachment and attachment of the nodes from and to the clusterheads.

The main objective of our approach is to cluster the network efficiently around a few high-energy clusterhead nodes. Clustering extends the life of the network by allowing the clusterheads to conserve energy through communication with closer nodes and by balancing the load among them. Since we assume that all nodes are identical and produce data at the same rate, to balance load in the system we have to balance the number of nodes in a cluster and the communication energy required per clusterhead.

Our next problem consists of proposing our new cluster degree constraint and our modified degree-difference schemes to replace those claimed in [5, 8, 10, 11, 12] where the authors claimed that according to their notation, the number of nodes that a clusterhead can handle ideally is bounded by a value v_i. For this purpose they computed for every node the *degree-difference* as follows:

$$\Delta_{v_i} = |deg\,(v_i) - \delta| \tag{8}$$

Unfortunately, here the authors did not explain how to select δ.

δ dictates that the node degree should be constrained and consequently has two conflicting consequences. Choosing little δ may lead to redundant number of clusters. In the other side, choosing high values for δ may decrease the number of clusters which leads to yields to clusterhead overloading and consequently to excessive energy consumption. To overcome these inefficiencies, we propose that an optimal δ value should be selected. This optimal value should be based on the historical statistical ad hoc mobility. Therefore, we strived to find a node degree constraint dependent to network size (*N*). In this context, we conducted an intensive research in the statistical results derived from well known works in literature, and we concluded that in general, the degree constraint verifies the following inequality:

$$\delta \leq 2ln\,(N) \tag{9}$$

The difference operator used in (8) cannot be considered as an efficient comparative tool. Consequently, next, we propose our analytical expression to solve this inefficiency observed in [5, 8, 10, 11, 12].

We propose to calculate the relative dissemination degree. This parameter reflects the relative deviation of the number of neighbors in a current setting from that ideal.

$$\beta(v_i) = \frac{|\delta - deg\,(v_i)|}{deg\,(v_i)} \tag{10}$$

3.3 Energy Consumption

It is known that more power is required to communicate to a larger distance. Therefore, we are motivated to evaluate the energy consumption. For this purpose, for every node v_i, we compute the sum of the distances, $D\,(v_i)$ with its neighbors, as:

$$D\,(v_i) = \sum_{j=1}^{n} dist\,(v_i, v_j) \tag{11}$$

3.4 Remaining Battery Energy

We have identified a weekness in WCA. It consists in computing the cumulative time during which a node acts as a clusterhead. This cannot guarantee a good assessment of energy consumption because data communication consumes a large amount of energy and varies greatly from node to node. Conseqently, we adopt a more simplified method. Each mobile node can easily estimate its remaining batterey energy $RBE\,(v_i)$. Consequently, a node with longer remaining battery lifetime is a better choice for a clusterhead.

4. SWCA Algorithm

Based on the preceding discussion, we propose an algorithm called Scalable Weighted Clustering Algorithm (SWCA) that effectively combines each of the above system parameters with certain weighting factors chosen according to the system needs [5]. The flexibility of changing the weight factors helps us apply our algorithm to various networks [5]. The output of clusterhead election procedure is a set of nodes called the dominant set. The clusterhead election procedure is invoked at the time of system activation and also when the current dominant set is unable to cover all the nodes. Every invocation of the election algorithm does not necessarily mean that all the clusterheads in the previous dominant set are replaced with the new ones. If a node detaches itself from its current clusterhead and attaches to another clusterhead then the involved clusterheads update their member list instead of invoking the election algorithm [3].

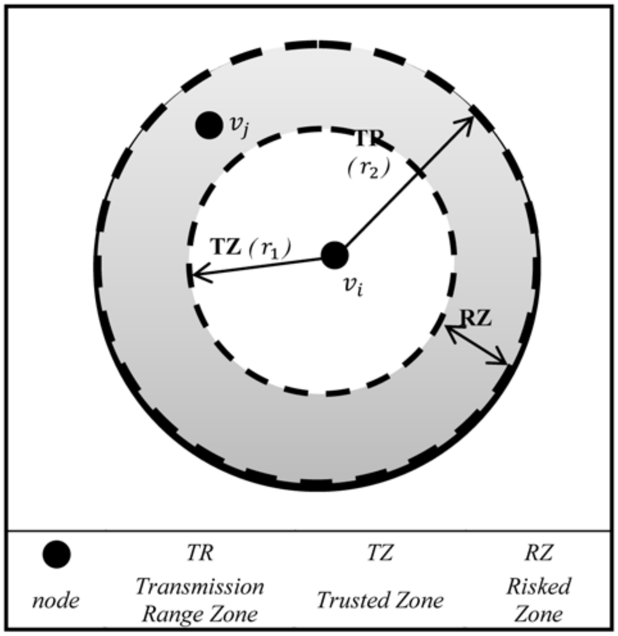

Figure 1. Transmission range zones

4.1 SWCA structure

Our algorithm is composed of two parts: clusterhead selection and formation of cluster members' set.

4.1.1 Cluster head selection

The cluster head selection process is composed of the following steps:

1. *Find the neighbors* (*degree*) *of each node* $\boldsymbol{v_i}$ *using* (2).
2. *For each node, calculate the stability factor using* (7).
3. *For each node, calculate the relative dissemination degree using* (10).
4. *Evaluate the energy consumption using* (11).
5. *Calculate the remaining battery energy of each node* (*RBE* (v_i)).
6. *Calculate the combined weight* $\boldsymbol{W}$ ($\boldsymbol{v_i}$) *for each for each node* $\boldsymbol{v_i}$: $\boldsymbol{W}(\boldsymbol{v_i}) = \boldsymbol{w_1 D}(\boldsymbol{v_i}) + \boldsymbol{w_2 RBE}(\boldsymbol{v_i}) + \boldsymbol{w_3 STF}(\boldsymbol{v_i}) + \boldsymbol{w_4 \beta}(\boldsymbol{v_i})$
7. *Select the node not situated on the border and having the minimum weight* $\boldsymbol{W}$ ($\boldsymbol{v_i}$) *as a cluster head.*
8. *Delete node* $\boldsymbol{v_i}$ *and all its* $\boldsymbol{N}$ ($\boldsymbol{v_i}$) *from G.*
9. *Repeat the* 7^{th} *and* 8^{th} *steps until G is empty.*

Here, w_1, w_2, w_3 and w_4 are the *weighing factors* for the corresponding system parameters and such that $w_1 + w_2 + w_3 + w_4 = 1$.

4.1.2 Cluster member formation

This stage constitutes the final step of our SWCA algorithm and represents the construction of the cluster members' set. Each clusterhead defines its neighbors at two hops maximum, which form the members of the cluster. In the following step, each cluster head stores all information about its members, and all nodes record the cluster head identifier. This exchange of information allows the routing protocol to function in the cluster and between the clusters. As the topology is dynamic, the nodes tend to move in different directions and at different speeds provoking clusters' configuration. Consequently, the position of the nodes and their speed must be updated periodically. The speed of a node is responsible for the change in its position. For this reason, the speed of the node generates the choice of the update time-slot [5]. Updates can be reduced by choosing longer time-slot, if the mobility of the node is low [5]. We should avoid periodical updates with higher frequency as they provoke great consumption of battery power and consequently increase the necessity of configuration changes [5].

4.2 Explanatory example

For a better comprehension of our algorithm, we take an example where the topology is arbitrary and the network is composed of 15 nodes (see figure 1(a)). The same original graph was used as a model on which the authors of [5] applied their *WCA* algorithm. This figure shows the initial configuration of the nodes in the network with individual node ids. Dotted circles with equal radius represent the fixed transmission range for each node. A node can hear broadcast beacons from the nodes which are within its transmission range. We demonstrate our SWCA algorithm with the help of figures 1 (b) and (c). An edge between two nodes in Figure 1 (b) signifies that the nodes are direct neighbors of each other.

Node #	Direct neighbor	$\deg(v_i)$	$stf(v_i)$	$\beta(v_i)$	$D(v_i)$	*Remen* (v_i)	$W(v_i)$
5	3, 4, 13	3	1.33	0.66	5	3	2.26
9	2, 10, 12, 14	4	3	0.25	5	5	2.42
8	7, 13	2	1.5	1.5	4	4	2.5
1	15, 10	2	1.5	1.5	6	2	2.9
15	1	1	1	4	3	0	3
7	6, 8	2	2.5	1.5	7	1	3.2
4	5	1	2	4	3	1	3.2
14	9, 12	2	3	1.5	7	1	3.25
6	7	1	2	4	3	2	3.3
11	10	1	1	4	3	3	3.3
12	9, 14	2	4	1.5	7	1	3.35
3	5	1	2	4	3	4	3.5
13	5, 8	2	2.5	1.5	7	4	3.5
2	9	1	2	4	4	3	3.7
10	2, 9, 11	3	3.66	0.66	12	4	4.7

Table 1. Execution of SWCA

All numeric values, are obtained from executing SWCA on the 15 nodes are tabulated in table 1, where the combined weight $W(v_i)$ is sorted in decreasing order. The degree $deg(v_i)$, which is the total number of neighbors a node has is shown in step 1. The stability factor is calculated in step 2 for each node. The dissemination degree for each node is calculated. This corresponds to step 3 in our algorithm. Thereafter, the energy consumption for each node is calculated in step 4. The remaining battery lifetime for each node is calculated as step 5 and in our table, these values are chosen randomly. After the values of all the components are identified, we compute the weighted metric, $W(v_i)$, for every node as proposed in step 6 in our algorithm. The weights considered are $w_1 = 0.3$, $w_2 = 0.5$, $w_3 = 0.1$ and $w_4 = 0.1$ Note that these weighing factors are chosen arbitrarily such that $w_1 + w_2 + w_3 + w_4 = 1$. We set $\delta \leq 2l\,n(N) = 2 \times \ln(15) \approx 5$. As seen from table 1, the nodes 5, 9, 8 and 1 are selected as clusterheads. The nodes 10 and 13 are selected as gateways.

The contribution of the individual components can be tuned by choosing the appropriate combination of the weighing factors [5]. Figure 1 (b) shows how a node with minimum $W(v_i)$ is selected as the clusterhead in a distributed fashion as stated in step 6 in our algorithm. The solid nodes represent the clusterheads elected for the network. Note that as a result of step 8, no two clusterheads are immediate neighbors. Figure 1(c) shows the initial clusters formed by execution of our SWCA clustering algorithm on the original graph depicted in Figure 1(a). Figure 1(d) shows the initial clusters formed by execution of WCA on the same original graph Figure 1(a). Although we kept the quarter of data used in Table 1 provided in [5] (node degree and remaining battery energy), it is obvious that the number of clusters generated by our algorithm (4 clusters) is lower than in WCA (8

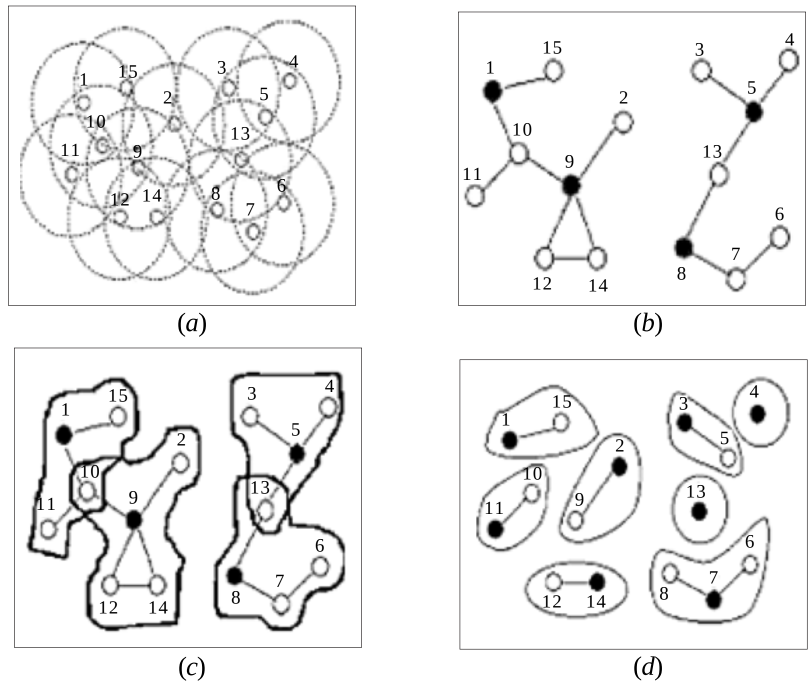

Figure 2. (a) system topology; (b) SWCA cluster head election stage;
(c) SWCA Cluster formation stage (d) WCA Cluster formation stage

clusters). This can be explained by the robustness of our parameters used to choose the clusterhead.

5. Conclusion

We have considered the problem of constructing a framework for dynamic organizing mobile nodes in wireless ad-hoc networks into clusters where it is necessary to provide robustness in the face of topological changes caused by node motion, node failure and node insertion/removal. Extending previous works, we have also mathematically derived a new clustering stability scheme. In the same objective We derived a simple clustering load balancing scheme. These two proposed schemes are considered as new mechanisms to overcome some inefficiencies detected in WCA and other similar clustering algorithms. It was shown that our proposed clustering algorithm performs similar to the best well-known algorithms (such as the WCA). The performance of our SWCA algorithm is proven by manual computation at this stage. However, we are now carrying a simulation based comparative study to validate the manual results.

References

[1] Pandi Selvam et al, R. (2011). Stable and Flexible Weight based Clustering Algorithm in Mobile Ad hoc Networks / (*IJCSIT*) *International Journal of Computer Science and Information Technologies*, 2 (2) 824-828.

[2] Pandi Selvam et al, R. (2011).Stable and Flexible Weight based Clustering Algorithm in Mobile Ad hoc Networks / (*IJCSIT*) *International Journal of Computer Science and Information Technologies*, 2 (2) 824-828.

[3] Suchismita Chinara, Santanu Kumar Rath. (2009). A Survey on One-Hop Clustering Algorithms in Mobile Ad Hoc Networks. *Journal Network System Management*; 17, 183–207.

[4] Naveen Chauhan et al. (2011). A Distributed Weighted Cluster Based Routing Protocol for MANETs. *Wireless Sensor Network*, 3, 54-60, doi:10.4236/wsn.2011.32006 Published Online February. (http://www.SciRP.org/journal/wsn)

[5] Chatterjee, M., Das, S. K., Turgut, D. (2002). WCA: a weighted clustering algorithm for mobile Ad Hoc networks. *Cluster Computing* 5, 193-204.

[6] Christian Bettstetter. (2004). The Cluster Density of a Distributed Clustering Algorithm in Ad Hoc Networks. *IEEE International Conference Communications*; 7, p. 4336 - 4340.

[7] Basagni, S. (1999). Distributed clustering for ad hoc networks. *In*: Proc. Intern. Symp. Parallel Architectures, Algorithms, and Networks (ISPAN); (Perth/Fremantle, Australia); June.

[8] Rathika, S. K. B., Bhavithra, J. (2012). An Efficient Fault Tolerance Quality of Service in Wireless Networks Using Weighted Clustering Algorithm; *Bonfring International Journal of Research in Communication Engineering*; 2 (1), Part 4, February.

[9] Tolba, Magoni, D., Lorenz. (2007). A stable clustering algorithm for highly mobile Ad Hoc networks, P. Second International Conference on Systems and Networks Communications. ICSNC 2007.

[10] Jin,et al. W. (2005). A load-balancing and energy-aware clustering algorithm in wireless ad-hoc networks; *Embedded and Ubiquitous Computing – EUC* 2005 *Workshops;* Lecture Notes in Computer Science, 3823, p. 1108-1117.

[11] Xi'an Jiaotong, et al. (2007). WACHM: Weight based adaptive clustering for large scale heterogeneous MANET; *Communications and Information Technologies*; ISCIT '07.

[12] Hui Cheng, et al. (2008). Stability-aware multi-metric clustering in mobile ad hoc networks with group mobility. *Wireless Communications and Mobile Computing*; 9:759–771, Published online 21 April in Wiley InterScience,

[13] Bollbas, B. (1985). Random Graphs; Academic Press.

[14] Amine Abid, M., Abdelfettah Belghith. (2011). Stability routing with constrained path length for improved routability in dynamic MANETs. The International Journal of Personnal and Ubiquitous Computing, Springer, 15 (8) 799-810.

[15] Gupta, G., Mohamed Younis. (2003). Load-balanced clustering of wireless sensor networks, *IEEE International Conference on Communications*, ICC '03, Anchorage, Alaska, USA, May 11-15.

[16] Brust, et al, M. R. (2007). WACA: A hierarchical weighted clustering algorithm optimized for mobile hybrid networks, *In*: Proceedings of the Third International Conference on Wireless and Mobile Communications ICWMC '07, Guadeloupe, French Caribbean, March 4-9.

[17] Abdelfettah Belghith, Jemili, I., Mosbah, M. (2007). A Distributed Clustering Algorithm without an Explicit Neighborhood Knowledge, *International Journal of Computing & Information Sciences* (*IJCIS*), 5 (1) 24-34, ISSN 1708-0479, April.

[18] Jemili, I., Abdelfettah Belghith, Mosbah, M. (2008). A Synchronous Tiered Based Clustering Algorithm for large-scale Ad hoc Networks, The 10th IFIP International Conference on Mobile and Wireless Communications Networks (MWCN'08), Toulouse, France, September 29-30.

[19] Jemili, I., Abdelfettah Belghith, Mohamed Mosbah. (2007). Algorithme distribué de clusterisation sans connaissance du voisinage: principe et évaluation, The 7th International Conference on New Technologies of Distributed Systems (NOTERE'07), Marrakech, Marocco, Juin 4-8.