Empirical Study of Clustering Algorithms for Wireless Ad Hoc Networks

C. Tselikis, S. Mitropoulos, C. Douligeris
Department of Informatics
University of Piraeus
80 Karaoli & Dimitriou Str., Piraeus 185 34, Greece
sarandis@unipi.gr

E. Ladis, K. Georgouleas, C. Vangelatos
Hellenic Aerospace Industry
Electronic Systems & Applications Design Dept.
P.O. Box 23, Schimatari 320 09, Greece
egladis@haicorp.gr

N. Komninos
Algorithms and Security Group
Athens Information Technology
Peania (Attiki), Greece
nkom@ait.edu

Abstract— **In this study we evaluate with experiments three generic clustering algorithms, namely the Lowest-ID, the Highest Degree and the Extended Robust Re-clustering Algorithm which is the one proposed. The aim is to investigate which are the factors that have significant effect on the re-clustering performance. We isolate those performance factors as being network conditions that we simulate with a particular focus on the node deployment pattern, the mobility pattern, the radio transmission range and the energy of the ad hoc nodes. For the evaluation of the re-clustering efficiency and for the comparison of the three algorithms we examined conventional re-clustering performance metrics, such as the cluster head modification rate and the number of the generated clusters but also reliability metrics, such as the cluster head availability probability and the end to end message delivery ratio. We draw generic outcomes that hold for the three algorithms and we also discuss the behavior of the proposed algorithm.**

Keywords: re-clustering, mobility, stabilitiy, reliability

I. INTRODUCTION

In mobile ad hoc networks (MANET) the nodes communicate in a peer to peer fashion without the necessity for a pre-existing network infrastructure (routers or other dedicated servers). Partitioning the MANET to a number of distinct clusters is a way to insert hierarchical levels into the network which results into assigning different roles to the nodes. The main roles are the cluster head role, the simple member role and the distributed gateway role, [3]. The cluster heads (CH) perform functions, such as aggregation of the intra-cluster messages, message encryption, inter-cluster message forwarding, radio resources and key management, device verification and user authentication and intrusion detection. The advantage in using hierarchical structures is that the higher layers offload and abstract the organization details of the lower in the hierarchy layers.

The problem can be that in dynamic networks the re-clustering procedure is initiated frequently thus imposing high communications overhead for the network and high battery consumption for the nodes. It is evident that the re-clustering should be efficient with as much less cluster head changes as possible. Considering the above, we investigate the factors (parameters) that affect a re-clustering algorithm's performance. For this reason, we examine the behavior of three basic clustering algorithms, namely the Enhanced Robust Re-clustering Algorithm (ERRA), the Lower-ID (LID) [2] and the Highest Degree (HD) [3] and we compare their performance behavior under various network conditions.

The paper is organized as follows. Section II states some of the most known re-clustering algorithms. Section III explains the motivation. Section IV states the Extended RRA (ERRA) algorithm in detail. Section V describes the factors and their models that we consider of importance for the re-clustering performance. Section VI presents the experimental results. Discussion and points for future work follow in section VII and section VIII respectively.

II. RELATED WORK

The Robust Re-clustering Algorithm (RRA) [1] is a graph-based weighted re-clustering algorithm that is based on two decision parameters, as it will be described in more detail later in section IV. The Lower ID (LID) algorithm [2] is one of the earlier static clustering algorithms. Being static, the LID assigns the cluster head role to the node with the lowest ID in a neighborhood. In case that a highly mobile node with a low ID

This work was partially supported by the University of Piraeus Research Centre.

roams into the network area, it will prevail in all the visited clusters and this will cause numerous updates with heavy overhead for the network and battery exhaustion for the node. One of the most popular topology-aware algorithms is the highest-degree algorithm [3] in which the node with the largest number of one-hop neighbors (i.e., the node with maximum connectivity degree) is elected as cluster head. Clustering with the highest degree is dictated by factors, such as the power levels of the nodes, the node mobility and their available battery. Improved schemes of the highest-degree algorithm include the Least Cluster Head Change algorithm (LCC, known to be a stable algorithm) [4] and the improved highest degree algorithm [5]. In the LCC if two cluster heads come in range the larger ID node is forced to give up his role – aiming to eliminate the radio interference –, while in [5] the algorithm elects cluster head the nodes that have many neighbors with small connectivity, an attempt to reduce the number of clusters. Among the stable re-clustering algorithms that are using the mobility metric are the DMAC [6] and the MOBIC [7], while one of the most known weighted schemes that combines system-wide parameters is the Weighted Clustering Algorithm (WCA) [8]. The LEACH cluster head selection algorithm [9], mainly applied to sensors networks, is the most representative algorithm of the energy-efficient family of clustering schemes. In [9] the sensors take autonomous decisions and randomization is introduced in order to spread uniformly to the network the total energy available.

III. MOTIVATION

When the re-clustering procedure is invoked frequently extra control overhead and extra management cost is introduced, only to set up short-living structures [10]. Therefore, the re-clustering procedure has to be efficient yielding the least possible communications cost, while keeping the advantage of creating self organized, scalable, autonomous networks. Which are the critical underlying network factors and in which way they make the re-clustering a more efficient procedure is the motivation for this study.

IV. EXTENDED ROBUST RE-CLUSTERING ALGORITHM

The Extended Robust Re-clustering Algorithm (ERRA) is an extension of [1] which is a deterministic, sequential, weighted-based re-clustering algorithm. ERRA is executed in two phases.

A. The Network Setup Phase

All the nodes are in the isolated mode i.e., the nodes neither hold the CH role, nor do they belong to any cluster as members. Then, while moving, it starts listening for neighboring nodes. If no neighbor is heard after a specified time interval, then the isolated node changes his state and claims itself CH node. Within a cluster a node can hold the role of the member or the cluster head exclusively.

B. The Cluster Head Selection Phase

For each candidate node *i* three decision parameters are examined by ERRA, namely:

1) *The connectivity degree* d_i i.e., the number of the one-hop neighbors in range of node *i*.

2) *The node's residual energy level,* E_{ri}. That is reduced whenever the node *i* elects as cluster head. Each node holds a counter $CH_{counter}$ to keep track of how many times it has been elected in the past.

3) *The distance of the node i from a reference point inside each cluster, which is denoted by* D_i. ERRA favors more those candidate nodes which are positioned closer to the center of the cluster.

The decision variable w_i that is calculated by each node *i* is the weighted sum of d_i, E_{ri} and D_i, as given in Equation (1).

$$w_i = a \times d_i + b \times E_{ri} + c \times D_i^{-1}, \tag{1}$$

where *a*, *b*, *c* are the decision coefficients satisfying:

$$\alpha + b + c = 1.$$

The coefficients *a*, *b* and *c* can be chosen depending on the topology of the ad hoc network and the requirements of the ad hoc application. For example, if we weight the battery coefficient *b* more than the connectivity degree coefficient *a*, ERRA will prefer for cluster heads those nodes that have more energy available, rather than the nodes that have more neighbors in range. ERRA breaks ties in favor of the lower ID node. The ERRA pseudo-code regarding the CH selection procedure is presented in Figure 1.

```
∀node ∈G {
    if (node ∉D) {
        build node_adjacent_list;
        CH = node with Max weight wi in node_adj_List;
        CH.setRole = cluster_head;
        CH.setCH(CH);
        CH_Table.add(CH);
        CH.Eres = CH.Eres - CH.Econsumed++;

        ∀(neigh ∈node_adj_List and neigh != CH) {
                    neigh.setRole(Member);
                    neigh.setCH(CH);
                    Member_Table.add(neigh);
                    D.add(neigh);
        }
        D.add(node);
        D.add(CH);
    }
}
```

Figure 1. The ERRA pseudo-code.

V. THE RE-CLUSTERING PERFORMANCE FACTORS

The factors that we expect to affect the re-clustering efficiency are the topology, mobility, energy consumption and the power level of the wireless ad hoc nodes. In this section we give short descriptions of the models that the above factors followed in the simulation experiments that we conducted.

A. Deployment Model

Recent studies attempt to quantify the relationship between the affinity parameters (like the degree, or other path-related parameters) and the robustness of complex networks, with some examples the works done in [13], [14], [15]. The mean connectivity degree of the nodes, i.e., the mean number of the one-hop neighbors that exist inside a node's range is a fundamental affinity property of a network topology.

Random graphs can be exploited for the study of the topology effect on the communications protocols and algorithms. Random graphs are generated by assuming various connectivity models with various probability distribution functions for the connectivity degrees. A known connectivity model that implicitly generates power laws for the node degrees is the Barabasi-Albert model [16] which is known as the 'preferential connectivity model': highly connected nodes become greedy by attracting more and more nodes. We created with the BRITE graph generator [11] two different deployment patterns of the ad hoc nodes with a mean average connectivity degree equal to 5. We used two network densities of 100 and 200 nodes for the same surface. Figure 2 shows 200 nodes which are deployed randomly in the field, i.e., the node degrees are distributed according to the negative exponential function.

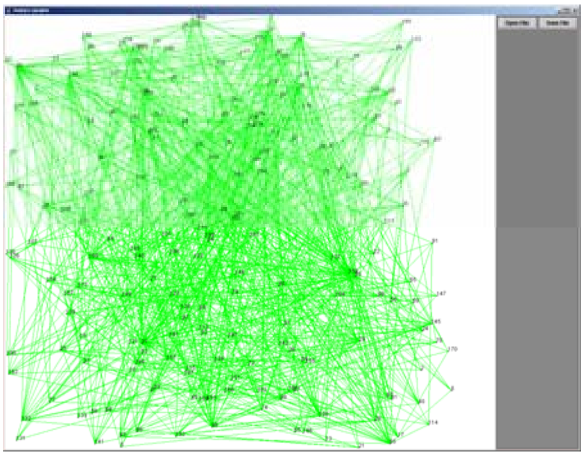

Figure 2. The random deployment pattern.

The second deployment pattern was based on the Barabasi-Albert model and followed power laws, more specifically the Heavy Tail Pareto distribution and it is shown in the Figure 3. Figure 3 illustrates the grouped ad hoc configuration with the initial concentrations of the 200 nodes clearly shown (note that the peripheral nodes in Figure 3 are less than those in Figure 2).

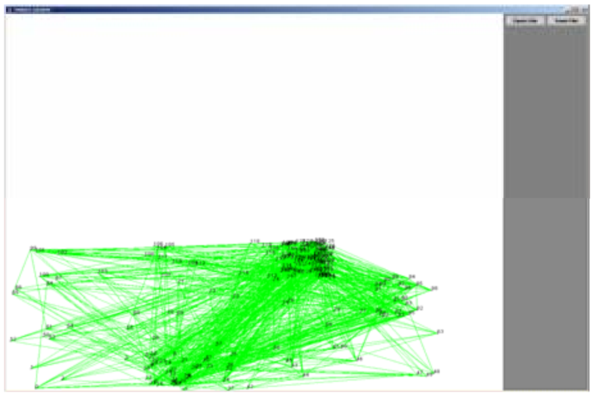

Figure 3. The grouped deployment pattern.

We simulated the ad hoc nodes as being the vertices of the above random graphs. On such nodes we attached our protocol stack. The stack includes the energy model, the mobility model, the IP packet handlers and on top of them the three re-clustering algorithms. Also, we used a dynamic addressing scheme, i.e., the graph identities (labels) of the cluster head nodes were used to derive the C-class address space which was allocated to the cluster member nodes.

B. Mobility Model

In our experiments mobility was simulated according to the Random Way Point (RWP) model. According to the random waypoint movement, a node chooses at random a destination point inside the bounds of the network area and with randomly chosen velocity travels towards this point following a straight path. When the node reaches the destination, pauses for a random time, randomly chooses a new destination point and the procedure repeats.

TABLE I. MOBILITY PARAMETERS

Parameter	***Description***	***Value***
RWP pause time	Index of mobility	0 – 40 seconds
RWP precision	Hop distance	10 meters
RWP speed	min – max speed	(5 – 50) km/h

In the experiments we assumed pausing times between 0 seconds, which corresponds to continuous movement, and 40 seconds, which corresponds to almost stationary nodes. The nodes traveled with maximum speed in the range between the pedestrian speed of 5km/sec and 50km/sec. The step precision was set to 10 meters, as shown in Table I.

C. Radio Power

When the nodes transmit in small radio power they cover small ranges and the network is partitioned to many cluster areas with minimum overlapping among them. When the radio covers larger areas a smaller number of significantly overlapping clusters with large membership is generated. It is worth mentioning that, for a given interference level, the optimum transmission range which maximizes the network throughput is different than the value of the radio range that maximizes the network connectivity level. According to [3], the radio range that maximizes the connectivity roughly infers a 0.4 reduction to the maximum of the network protocol throughput.

VI. Experimental Results

The metrics that we evaluated through simulation experiments for the three compared re-clustering algorithms are:

1) Re-clustering stability. It is the main clustering performance metric which gives the average per node number of cluster head changes. Obviously the less stable an algorithm is the more the information updates that are needed and hence the more the overhead imposed to the network communications. Selection (Heading 4): Highlight all author and affiliation lines.

2) Number of clusters. The smaller the number of clusters generated by the clustering algorithms the better for the routing protocol, since the size of the routing tables is preserved small and routing calculations become lighter.

3) End to end message delivery. A metric to test the efficiency of the re-clustering algorithm is to evaluate the application-level messages that actually reach their destination across robust routes.

We used a discrete-event simulator, namely the Java Network Simulator (JNS), [12]. As input to the JNS we used the topology scripts that we generated incrementally by using the BRITE topology generator [11]. JNS parses the topology scripts and is also extended to support the Random Waypoint Mobilty and the transmission patterns that are shown in Table II. Each result point is the mean value of three experiments.

TABLE II. Transmission Parameters

Parameter	*Description*	*Value*
Transmission Range	Radio range	(0.5 – 200) m
Packet size	Network Data Unit	512 Bytes
Flooding rate	TCP/UDP packet transmission rate	100pks/sec

A. Re-Clustering Stability

Figure 4 illustrates results for the cluster head change rate in experiment #1. Experiment #1 simulates a dense network of 200 nodes with users moving with the very high speed of 50km/h. Mobility was simulated with the random waypoint model. It is clear in Figure 4 that for in the dense network scenario ERRA outperformed both HD and LID, especially for the ranges between 10meters<radius<200meters, for both the grouped and the uniform initial deployment. In the small ranges, below 10 meters, HD performed better whith grouped placement rather than with uniform placement. On the contrary, ERRA performed better when using the random placement model, notably across the whole range of the radio radii. Also Figure 4 shows that LID preserved superiority against HD and ERRA only for ranges up to one meter. For larger ranges and power the LID stability experienced severe degradation yielding a large number of cluster head changes, especially in the medium ranges between 10m and 100m.

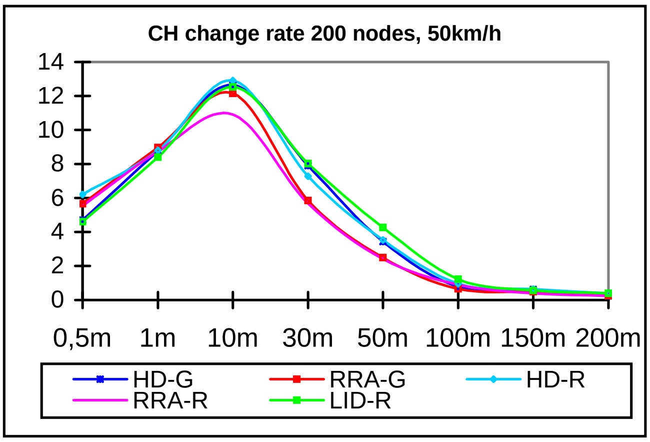

Figure 4. Comparison of CH modificaton rate in Experiment 1.

Figure 5 illustrates results for the cluster head modification rate in Experiment #2. Experiment #2 corresponds to a sparse network of 100 nodes and wireless nodes that move with high average speed (50km/h).

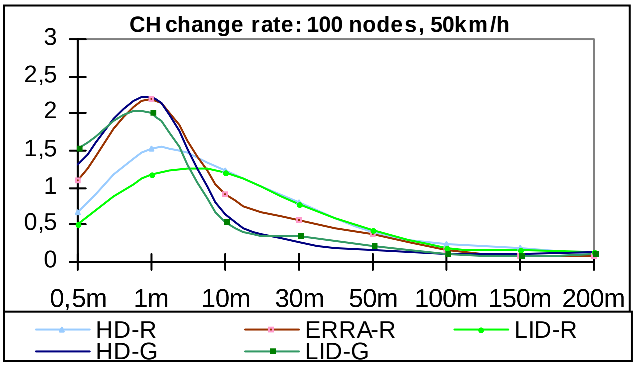

Figure 5. Comparison of CH modificaton rate in Experiment 2.

Figure 5 shows that in the radio ranges less than one meter the most stable re-clustering of the three compared algorithms was the LID, a fact also observed in Figure 4 When increasing the radio coverage the most stable re-clustering was achieved with HD and ERRA re-clustering. Especially, Figure 5 illustrates that for small radio ranges the performance is better with random deployment rather than with grouped deployment, while for large radio ranges the opposite holds. When the nodes are capable of large power the change rates are reduced when adopting the grouped node deployment strategy, a fact also that was also observed in Figure 4.

B. Number of Clusters

Figure 6 illustrates results regarding the number of the generated clusters after applying the three algorithms. It is shown in Figure 6 that the algorithms created fewer clusters when the nodes were deployed in groups rather than when the nodes were placed randomly. ERRA created a sufficiently small number of clusters in the case of radio ranges of more than 30 meters.

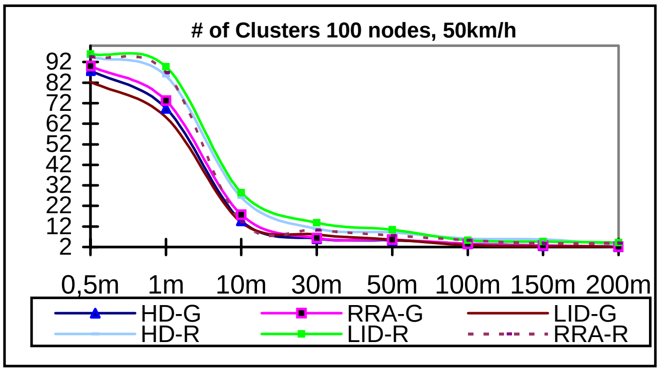

Figure 6. Comparison if CH modificaton rate in Experiment 2.

Figure 7 shows the improving effect on the number of clusters which were generated when introducing the third weighted parameter *Dm* in the Robust Reclustering Algorithm. The parameter *Dm* is defined as the node's distance from the cluster center, see Equation (1).

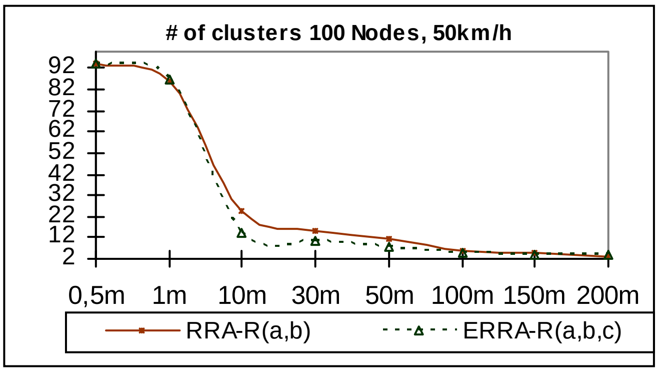

Figure 7. ERRA (a, b, c) vs. RRA (a, b).

Figure 7 illustrates that when ERRA takes into account three parameters, rather than two, a smaller number of clusters is created. The reason is that with the addition of the parameter *Dm* the allocations of the CH role to nodes wandering at the network perimeter are avoided. Choosing nodes at the bounds of the field as cluster heads renders the total hierarchy ineffective because of the unsustainable large number of clusters and this is avoided by ERRA.

C. Message Reliability

The reliability metric that we used to estimate the impact of the re-clustering algorithms on the ad hoc communications is defined in Equation (2):

$$\text{Reliability} = 1\text{- MEAN}\left(\frac{\text{\#of droppedmessages}}{\text{total\#of messagesent}}\right) \qquad (2)$$

The message reliability percentages were calculated after simulating a mixed traffic of the connection-oriented Simple GoBackN (SGN) protocol and the connection-less UDP transport protocol. Interesting is how the deployment model affects the percentage of the reliably delivered messages. Figure 8 illustrates that in the case of 200 nodes moving with average speed of 5km/hour all three algorithms, remarkably ERRA the most, increased their reliability in terms of the successfully transferred messages when switching the initial configuration from the random deployment to the grouped one.

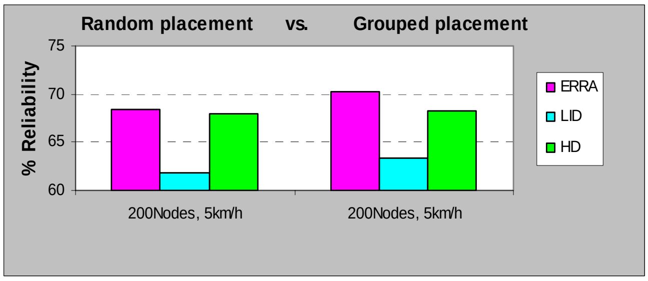

Figure 8. Comparison of the message reliability metric.

VII. DISCUSSION

The generic outcome that holds for the three algorithms is that the re-clustering performance is sensitive to such factors as the initial deployment pattern, the network density, the nodes' mobility model, and the radio transmission power level. More specifically:

- The instability point when re-clustering in the dense network case was shifted 10 meters to the right of the radio range scale when compared to the instability point of the sparse network. This is due to the fact that as the number of nodes increases, new visiting nodes enter the cluster areas. In effect, in the case of dense networks to achieve the same level of re-clustering stability the wireless nodes have to transmit with more radio power.
- Different node placement models (for example random and heavy tail distributions) require different network densities in order to achieve the same level of re-clustering stability under the same radio power.
- The topology that optimized the re-clustering stability of the three examined algorithms was shown to be the random (uniform) for the small radio power levels, while for larger power level the grouped pattern allowed to the re-clustering algorithms to incur less cluster head changes and therefore less overhead.

VIII. CONCLUSIONS AND FUTURE WORK

The conclusions that we draw regarding the ERRA performance are:

- In the experiments the ERRA was found more stable in the case of large radio ranges rather than in the case of small radio ranges. This behavior of ERRA is justified by the choice of the coefficients *a, b* and *c*, see Equation (1). In more detail, in the experiments we favored as for cluster heads those nodes having larger degree d_i rather than those nodes with more energy left, by choosing a larger value for the degree coefficient α than the energy coefficient *b*. This choice favored the powerful nodes and hence ERRA generated less cluster head changes in the case that the network is dense having nodes with large connectivity degrees (it might be a battlefield ad hoc scenario). In the sparse network case (it might be a home ad hoc application) where energy is of more importance than connectivity, ERRA degraded in stability with the above parameter setting.

- ERRA delivered the application-level messages to the final destinations more reliably than both the Lower ID and the Highest Degree. ERRA also gave better cluster head availability performance, an index of robustness against the event of random phenomena.
- In this paper we compared three clustering algorithms and we proposed an improved version of the RRA [1] for the wireless mobile ad hoc network. For our testing purposes we experimented with simulation of the random waypoint model. It is interesting to consider additional mobility models, such as the random walk model, the motorway model, or the movement of nodes in groups and then to evaluate the mobility impact over the re-clustering performance metrics.
- Moreover, the investigation of how the various end-user applications that produce different traffic patterns than those examined here might affect the performance of the underlying re-clustering schemes remains an open issue. The future work will also address the problem of securing the clustering procedure from malicious intruders that might threaten the availability of the cluster head nodes.

ACKNOWLEDGMENT

The work presented in this paper was partially supported by the EU-funded FP7 211998 AWISSENET project.

REFERENCES

[1] C. Tselikis, C. Douligeris, S. Mitropoulos and N. Komninos. *"Consistent reclustering in Mobile Ad Hoc Networks"*, IEEE Proceedings of the IWCMC 2008, pp. 825-830, Crete Island, 6-8 August 2008.

[2] Ephremides, A., Anthony, J., E., Baker, D., J. *A design concept for reliable mobile radio networks with frequency hopping signalling*. Proc. IEEE 1987; vol. 75, no. 1, Jan. 1987, pp 56-73.

[3] Gerla, M., Tsai, JTC. *Multicluster, mobile, multimedia radio network*. ACM-Baltzer J Wireless Networks, 1995; vol. 1, pp. 255-265.

[4] Chiang C. C., Wu, H. K., Liu, W., Gerla, M. *Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel*. Proc IEEE SICON, Singapore, 1996.

[5] Taniguchi, H., Inoue, M., Masuzawa, T., and Fujiwara, H. *Clustering Algorithms in Ad Hoc Networks*. Electronics and Communications in Japan, Part 2, Vol. 88, No. 1, 2005, pp 51-59.

[6] Basagni, S. *Distributed clustering for ad hoc networks*. Proceedings of the International Symposium on Parallel Architectures, Algorithms and Networks, (I-SPAN'99), Perth/Fremantle. June 1999, pp. 310-315.

[7] Basu, P., Kham N., Little, T. D. C. *A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks.* International Conference on Distributed Computing Systems Workshop, Apr. 2001.

[8] Chatterjee M., Das S. K., Turgut D., "WCA: A Weighted Clustering Algorithm for Mobile Ad Hoc Networks", Cluster Computing Journal, vol. 5, no. 2, Apr. 2002, pp. 193-204.

[9] Chandrakasan, A., Heinzelman, W., R., Balakrishnan, H. *Energy-efficient communication protocol for wireless micro sensor networks*. In: 33rd annual Hawaii International Conference on System Sciences, HICSS, page 30053014, 2000.

[10] Akyildiz, I. F., Wang, X. A Survey on Wireless Mesh Networks. IEEE Radio Communications, September 2005, pp. 23-30.

[11] Medina, A., Matta, I., and Byers, J. *BRITE: A flexible generator of internet topologies*. Technical Report 2000-005, CS Department, Boston University, Jan. 21, 2000.

[12] Java Network Simulator (JNS), <http://jns.sourceforge.net>.

[13] Lucas, G., Ghose, A., Chuang, J. *On Characterizing Affinity and its Impact on Network Performance*. Proceedings of the ACM SIGCOMM 2003 WORKSHOPS, Karlsruhe, Germany, August 2003, pp 65-75.

[14] Réka, A., Hawoong, J. & Albert-László, Barabasi. *Error and attack tolerance of complex networks*. Nature, vol. 406, 2000, pp. 378-381.

[15] Tilak, S., Abu-Ghazaleh, N., B., and Heinzelman, W. *Infrastructure Tradeoffs for Sensor Networks.* WSNA'02, Atlanta, Georgia, USA, September 28, pp 49-58.

[16] Barabasi, A., L., Albert, R. *Emergence of scaling in random networks*. Science, 286(5439), Oct. 1999, pp. 509-512.