以效能考量之ad-hoc式無線網路群集演算法

李棟村

國立宜蘭技術學院電子工程系講師

摘 要

本文考慮能量消耗、平衡負載及較好之骨幹之形成來考量ad hoc 式無線網路的群集問題，能量消耗於群集首(clusterhead)及其成員間為選群集首主要考量點，組員必須經常地將其狀況與群集首間做溝通，如此群集首方能得到最新訊息且群集內之路由及訊息之傳遞亦較能有效達成；再者，考慮群集過程之平衡負載.，一搬來說一無線網路節點只配有限電量，所以期望大部分的節點均有機會可當作群集首且每一群集能平均分配成員；其次，如果群集首能位於由節點及線段所形成之圖(graph)之內部，如此將會形成較好之骨幹來，主要原因是如果群集是由很多位於周圍之單一節點組成的話，由群集首及閘道相互合作而成之路由工作會變的較沒有效率.此篇提出一簡單演算法考慮這些因素，由模擬得知可確實達成此目標。

關鍵詞：群集技術、ad hoc 網路、多次跳達無線網路

An Energy-efficient Clustering Algorithm for Wireless Ad Hoc Networks

Tung-Tsun Lee

Lecturer, Department of Electronic Engineering, National Ilan Institute of Technology

Abstract

In this paper, we try to cluster nodes in ad hoc networks considering of energy consumption, load-balanced and well-backboned factors. The energy spent in communication between clusterhead and its members are considered to be the major factor in clusterhead election. The member of a clusterhead has to report its position with respect to its clusterhead in a timely fashion, such that the clusterhead can get the fresh information. And, the routing and message delivering can be done within cluster effectively. Furthermore, the load-balanced factor is considered in clustering process. A mobile node is equipped with limited power in general; therefore, it is desirable if most of nodes have the chance to be clusterhead, and most of clusters contain average number of members. Moreover, the well-backboned architecture formed by cluster is beneficial if clusterheads are within the inner of the graph derived from nodes and edges between communicable nodes. The reason is that routing process is achieved by the cooperation of clusterheads and gateway nodes; the routing activity is less effective if cluster is consisted of a single node located at boundary of graph. A simple algorithm considered upon these factors is proposed. The objectives can be achieved from the observation of simulations.

Key Words：clustering technique, ad hoc network, multihop wireless network

I. Introduction

Multi-hop mobile (or ad hoc) wireless networks can rapidly be deployed without the support of fixed infrastructures. Each node in the network works as a router as well as a host. Nodes are free to move randomly and organize themselves arbitrarily. Thus, the topology of the network may change rapidly. These networks are found in applications in military operations, disaster relief, and short-term activities.

A major challenge in multihop, multimedia networks is the ability to account for resources so that bandwidth reservations can be placed on them. We note that in cellular (single hop) networks such accountability is made easy by the fact that all stations learn of each other's requirements, either directly, or through a base station in cellular systems. This solution can be extended to multihop networks by creating clusters of radios, in such a way that access can be controlled and bandwidth can be allocated in each cluster. The notion of cluster has been used also in earlier packet radio nets, but mainly for hierarchical routing rather than for resource allocation.

Clustering is a technique to group ad hoc nodes into entities called clusters. There are several advantages in this logical relationship. Firstly, it is sufficient for only the nodes in its clusters to update their topology information, not all in this system. That makes fewer overheads for tracking ad hoc nodes. Secondly, the routing table in each node can be reduced. Besides, the generation and propagation of routing information can be scaled down. Recently, several researchers [1-10], have addressed in segmenting the whole network into individual clusters.

The primary step in clustering is the election of nodes called clusterheads and the formation of clusters with corresponding members. Several heuristics have been proposed to choose clusterheads in ad hoc networks. These include (i) Degree-based (ii) ID-based, and (iii) Weight-based. In Degree-based [2], the node with the maximum degree is chosen to be a clusterhead. The neighbors of cluster-head become members of that clusterhead and can no longer participate in the election process. In ID-based [1] [3], Gerla and Tsai proposed a simple heuristic by assigning a unique id to each node and choosing the node with the minimum id as clusterhead. The drawback of this heuristic is smaller id which may lead to battery drainage. In Weight-based [4], they assign node weight based on some criteria. A node is chosen to be clusterhead if its weight is higher than any of its neighbors.

In this paper, we try to cluster nodes in ad hoc networks from the following point of views. Firstly, the energy spent in communication between clusterhead and its member is considered to be the major factor in clusterhead election. The member of a clusterhead has to report its position with respect to its clusterhead in a timely fashion, such that the clusterhead can get the fresh information. And, the routing and message delivering can be done within cluster. Secondly, the **load-balanced** feature is considered in clustering process. A mobile node is equipped with limited power in general; it is desirable if most of nodes have the chance to be clusterhead, and most of clusters contain average of members. Finally, the **backbone of architecture** formed by clusterhead is better within inner of graph derived from the nodes and edge between communicable nodes. The routing process is achieved by the cooperation of clusterheads and gateway nodes; the routing activity is less effective if cluster is consisted of a single node located at boundary of graph.

The remainder of the paper is organized as follows. In session II, we give the overview of clustering techniques. In session III, the energy-efficient clustering algorithm is depicted. In session IV, simulation result and comparison are shown. Finally, in session V, we conclude with brief discussion.

II. Previous Work

In previous work, several heuristics have been proposed to choose clusterheads in ad hoc network. These include (i) Degree-based heuristic (ii) ID-based heuristic and (iii) Weight-based heuristic. In the simplified graph model of the network, the mobile terminals are represented as nodes and there exists an edge between two nodes if they lie within the transmission range of each other. We summarize these heuristics below.

A. Degree-based heuristic

This approach considers the node with maximum degree is chosen to be a clusterhead and any tie is broken by the node ids, which are unique. The neighbors of a clusterhead become members of that cluster and can no longer participate in the election process. This heuristic is also known as the highest-connectivity algorithm. Experiments demonstrate that the system has a low rate of clusterhead changes but the throughput of the system is low. Typically, each cluster was assigned some resources, which was shared among the members of that cluster on a round-robin basis [3]. As the number of nodes in a cluster is increased, the throughput of each user drops and hence, a gradual degradation in the system performance is observed. This is the inherent drawback of the Degree-based heuristic since the number of nodes in a cluster is not bounded.

B. ID-based heuristic

Gerla and Tsai [3] proposed a simple heuristic by assigning a unique id to each node and choosing the node with the minimum id as a clusterhead. Furthermore, the clusterhead can delegate its duties to the next node with the minimum id in its cluster. For the purpose of routing, a node is called *gateway* if it lies within the transmission range of two or more clusters. The routing process can be achieved by the cooperation of clusterheads and gateways. Upon this heuristic, the system performance is better compared to the Degree-based heuristic in terms of the clusterhead duration and member duration. Since the environment under consideration is mobile, it is unlikely that node degrees remain stable resulting in frequent clusterhead updates. The drawback of this heuristic is its bias towards nodes with smaller id which leads to the battery drainage of certain nodes.

C. Weight-based heuristic

Chatterjee et al. [4] assigned *node-weights* based on the some parameters of a node being a clusterhead. A node is chosen to be a clusterhead if its node-weight is higher than any of its neighbor' s node weights. The smaller node id is chosen to break a tie. To verify the performance of the system [4], the nodes were assigned combined weights on degree, transmission power, mobility and battery power. Results proved that the number of updates required is smaller than the Degree-based and ID-based heuristics. Since node weights were varied in each simulation cycle, computing the clusterheads becomes very expensive and there are no optimizations on the system parameters.

III. Clustering Algorithm

Few of the above heuristics leads to an optimal selection of clusterheads since each deal with only a subset of parameters that impose constraints on the system. For example, a clusterhead may not be able to handle a large number of nodes due to resource limitations even if these nodes are its neighbors and lie well within its transmission range. Thus, the load handling capacity of the clusterhead puts an upper bound on the node-degree. In other words, simply covering the area with the minimum number of clusterheads will put more burdens on the clusterheads. At the same time, more clusterheads will lead to a computationally expensive system. This may result in good throughput, but the data packets have to go through multiple hops resulting in high latency. In summary, it is still an important problem to choose an optimal number of clusterheads to yield high throughput but incur as low latency as possible. As the search for better heuristics for this problem continues, we try to consider property of the clusters after election process. We hope that the maintenance power between clusterhead and its member can be reduced, the variance of cluster size consisted of members can be kept lower, and the backbone formed by clusterheads and its member can be lied within inner of groups of nodes. It is evident that the algorithm should be executed distributed upon the observation from neighbors.

A. Basics for the Algorithm

To be noted that we consider one hop distance between clusterhead and its members. To decide how well a node suited to be a clusterhead, we try to cluster nodes in ad hoc networks from the following points of view:

Energy Consumption

The clusters can be established after the clusterhead election procedure executed. The **energy** is required in communication between clusterhead and its member for reason to keep the precise relationship between clusterhead and its members. Besides, the routing and message delivering processes have to be progressed among clusters continuously.

Load-Balanced

It is unwanted if the variance of cluster size is high. The **load-balanced** factor is important in clustering process. A mobile node is equipped with limited power in general, and it is desirable if most of nodes have the same chance to be clusterhead. Moreover, most of clusters can contain average number of members. That makes the power consumption in routing and message delivering processes to be better distributed.

Well-backboned

The **backbone of architecture** is formed by clusterheads and gateways. The routing procedure will forward packets on the backbone. It's not desirable if a cluster consists of a clusterhead alone, or if the clusterhead is located at the boundary of a cluster. Therefore, clusterhead had better within inner of graph derived from the nodes and edge between communicable nodes. Thus, routing activity can be more effective.

B. Proposed Algorithm

We are given a set *V* of *n* nodes. A node is able to send message with arbitrary power *p*. Nodes can vary their power, but not beyond a maximum power *P*, that is $0 \le p \le P$. We assume the existence of an underlying MAC layer that resolves inference problems. If node *u* broadcasts with power *p*, the node that can receive node *u*'s broadcast message (the set *N*) will acknowledge (with another broadcast message) to node *u*. After having received acknowledge of all nodes in *N*, node *u* knows the set *N* and corresponding transmission power between *u* and its neighbors.

Definition 1: The *least p*ower function $l(u, v)$ gives the minimum power for node *u* required to communicate its neighbor node *v*.

Definition 2: Given a multihop wireless network W=(M,L), where M is the number of nodes and L is the location of these nodes. The estimated avera*ge energy* $E(u)=\sum_{i=1}^{k} l(u,v_i)/k$ gives the average power for node u to communicate with its neighbors, where v_i is the neighbor of node u.

The energy-efficient clusterhead election process (**ECEP**) is shown in the following. Firstly, the boundary of graph is processed at beginning (step 3 of ECEP). The purpose of the preprocess step is to achieve the objective of *well-backboned* architecture. The isolated clusterhead can be avoidable and clusterhead can be located at inner of a group of nodes. Furthermore, the preprocess step will be advantageous to the objective of *load-balanced* property. The averaged size of cluster can be formed. Secondly, the energy for clustering is compared. In step 6, a node will join to existing neighbor with lowest transmission power with respect to $node_i$, which has declared to be a clusterhead. The reason of this step is to keep the number of clusterhead lower. After that, step 8 will check the possibility to be clusterhead if its $E(node_i)$ is lowest compared with its neighbors. This considers the energy-efficient factor for clustering problem.

Energy-efficient Clusterhead Election Process (ECEP) for $node_i$

```
1. if node_i is isolated    //no neighbor
2.     Declare node_i to be clusterhead
3. else if any neighbor with degree is equal to 1    //boundary of graph
4.     if (degree of node_i > 1) or (id of node_i > id of the neighbor)   //check the possibility of one-to-one
4.     if (degree of node_i > 1) or (id of node_i > id of the neighbor)   //check the possibility of one-to-one
5.         Declare node_i to be clusterhead
6. else if any neighbor declares to be clusterhead    //to avoid clusterhead explosion
7.     Join to node_j with lowest l(node_i, node_j) within these neighbors
8. else if E(node_i) is lowest compared with neighbors   //energy-efficient consideration
9.     Declare node_i to be clusterhead
```

In order to make decision on clusterhead election, the information of following is required for broadcasting to neighbors. The time complexity and the message complexity is *O (N)*. It's depending on the number of neighbors.

Source Node ID	Estimated Average Energy	No of Neighbor

D. Illustrative Example

In order to show the concept of ECEP, a simple example shown in Fig. 1 is illustrated. ECEP is executed at each node in distributed way. At beginning, node 1 and node 9 will declare to be clusterhead, since node 1 connects to a boundary node 4(and node 9 connects to a boundary node 3). Furthermore, node 5 and node 8 will join to the cluster of node 1, and node 2 will join to cluster of node 9 respectively. At the same time, node 0 declares itself to be clusterhead since its estimated average energy is smallest compared with node 6 and node 7. Node 6 and node 7 will join to node 0 after receiving clustering message from node 0.

IV. Simulation

We simulate a system with N nodes on a 200x200 unit area. Two nodes are said to have a wireless link between them if they are within communication range of each other. The performances are simulated with communication range of nodes set to 20 and 30 units. We assumed systems running with 100, 200, 300 and 400 nodes to simulate ad hoc networks with varying of node density. Also, the maximum hops between nodes with its clusterhead were set one. Some of noteworthy statistics are measured: ***Cluster Size Distribution, clusterhead Statistic, Energy Consumption, Number of Clusterhead, Clusterhead Duration,*** and ***Member Duration***. The *Cluster Size Distribution* shows the cluster consisted of clusterhead and its members. It can show the *load-balanced* and *well-backboned* features depicted in previous. The *clusterhead Statistic* show the statistic of ever being clusterhead. The *Energy Consumption* sums the power of clusterhead for broadcasting and the power of member for reporting to its clusterhead. The *Number of Clusterhead* can't be too large to avoid that each clusterhead manages few members. On the other hand, it can't be too few clusterheads that will be overloaded for clusterhead. The *Clusterhead Duration* is the mean time for which once a node is elected as a clusterhead. The longer of the duration makes more stable for the system. The *Cluster Size* is the mean size of cluster. The value is related to *Number of Clusterhead* and can't be too large and small. The *Cluster Member Duration* is the mean time a node stays a member of a cluster before moving to another cluster. This statistic is measure of stability like *Clusterhead Duratio*n. The *Size Distribution* shows the distribution cluster size in system running with specific number of nodes.

We compare ECEP with Degree-based and ID-based by following scenarios of mobility situations, where 'one-step markov path model' is shown in Fig. 2:

1. A: The speed of nodes are no more than 1/2 of wireless communication range with one-step markov path model .
2. A+: The speed of nodes is no more than 3/4 of wireless communication range with one-step markov path model. The speed of nodes is higher than in scenario A. This causes cluster head duration and member duration lower in general.
3. A-: The speed of nodes are no more than 1/4 of wireless communication range with one-step markov path model .The speed of nodes are lower than in scenario A and A+.
4. B:The speed of nodes are no more than 1/2 of wireless communication range(same as scenario A) with same probability in eight directions, instead of one-step markov path model . i.e. the movement is memoryless.

The most intuitive performance comparison of Energy-efficient, Degree-based and Id-based heuristics are ***Cluster Size Distribution, Clusterhead Statistic*** and ***Energy Consumption***. Fig. 3 shows ECEP with more of cluster size closed to normal distribution. Especially, degree-based and Id-based heuristic will make cluster size consisted of single node or too many of members. That is to say, ECEP can achieve the objectives of load-balanced and well-backboned features. Fig. 4 shows ECEP with less energy to maintain cluster structure. In order to show the load-balanced feature, Fig. 5 shows statistic of nodes ever being a clusterhead. Id-based gives the highest priority to lowest id, and make the highest load to these nodes. ECEP give all nodes have the same chance to be clusterhead. Fig. 6, 7 and 8 show the other properties that are listed for comparison in general for clustering algorithms. Id-based get the highest performance in clusterhead duration and member duration, since it elected clusterhead based on node id.

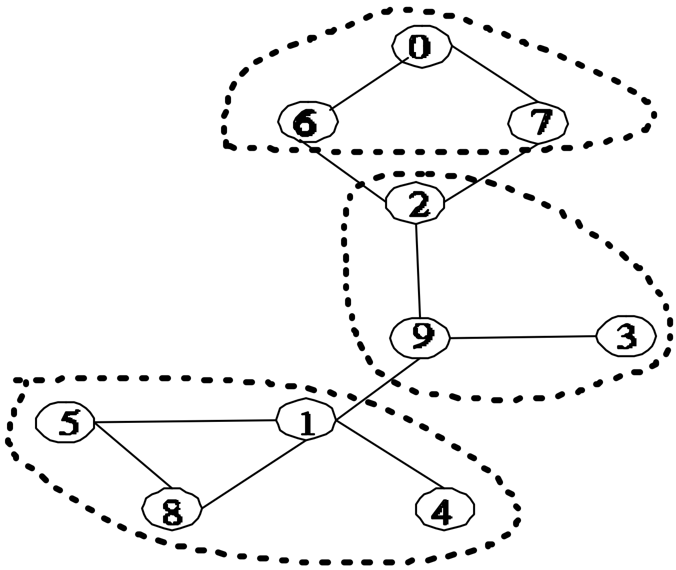
V. Conclusion

We have proposed a simple clustering algorithm in ad hoc networks considering of energy consumption, load-balanced and well-backboned features. We concentrate on the relationship of cluster and routing. The routing process is achieved by the cooperation of clusterheads and gateway nodes; the routing activity is less effective if cluster is consisted of a single node located at boundary of graph. A simple algorithm considered on load-balanced and well-backboned property is proposed. Especially, the energy spent in communication between clusterhead and its member is considered to be the major factor in clusterhead election. The member of a clusterhead has to report its position with respect to its clusterhead in a timely fashion, such that the clusterhead can get the fresh information. And, the routing and message delivering can be done within cluster. The objectives can really be achieved from the observation of simulations.

References

1. D. Amis, R. Prakash, T. H.P. Vuong and D. T. Huynh, (2000), "Max-min d-cluster formation in wireless ad hoc networks," *Proc.IEEE Infocom*.
2. K. Parekh, (1994), "Selecting routers in ad-hoc wireless networks", *ITS*.
3. R. Lin and M. Gerla, (Sep. 1997), "Adaptive clustering for mobile wireless networks", *IEEE J. Select. Areas in Commun.* Vol. 15, NO. 7.
4. Mainak Chatterjee,Sajal K. Das and Damla Turgut, (2000), "An on-demand weighted clustering algorithm(WCA) for ad hoc networks," *Proc. IEEE Globecom*.
5. Prithwish Basu, Naved Khan and Thomas D.C. Little, (2001) , "A mobility based metric for clustering in mobile ad hoc networks," *Distributed Computing System Workshop*.
6. X. Hong, M.Gerla, G. Pei, and C. C. Chiang, (1999) "A group mobility model for ad hoc wireless networks," *Proc. ACM/IEEE MSWiM '9*9.
7. Kim, Seokjae Ha, and Y. Choi, (1998), "K-hop cluster-based dynamic source routing in wireless ad-hoc packet radio network," *Proc. VTC* .
8. Suman Banerjee, Samir Khuller, (2001), "A clustering scheme for hierarchical control in multi-hop wireless networks," *Proc. IEEE Infocom*.
9. Ting-Chao Hou and Tzu-Jane Tsai, (Sep. 2001), "An access-based clustering protocol for multihop wireless ad hoc networks," *IEEE J. Selected Areas in Commun.* Vol. 19, NO. 7.
10. P.Johansson,T.Larsson, N.Hedman,B.Mielczarek and M.Degermark, (Aug. 1999), "Scenario-based performance analysis of routing pro-rocols for mobile ad hoc networks," *Proc. ACM Mobicom*, Seattle WA.

91 年 08 月 27 日投稿
91 年 09 月 20 日接受



Node Id	Energy
0	18
1	50
2	48
3	16
4	16
5	25
6	41
7	41
8	18
9	41

Fig 1　An Example illustrates the concept of ECEP.

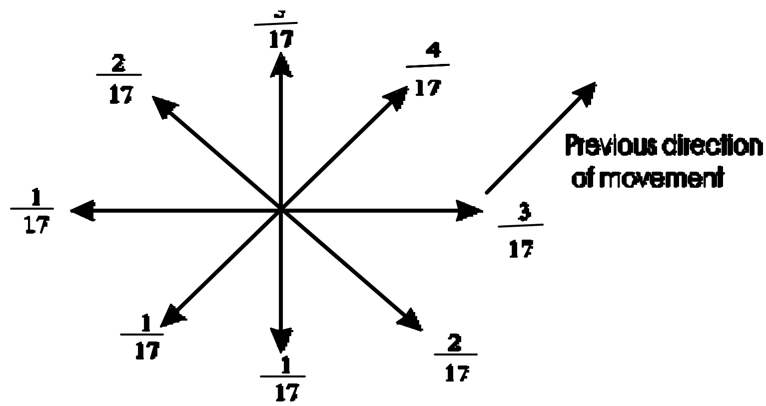


Fig 2　One-step markov path model with memory characteristic, higher priority to previous moving direction.

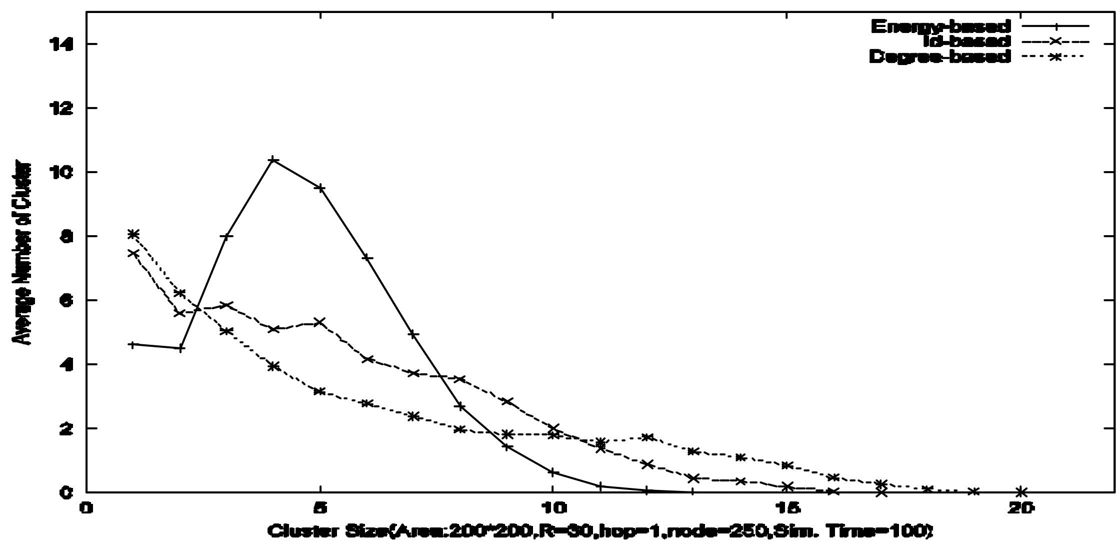


Fig 3　The average number of cluster versus cluster size in 250 nodes. ECEP gives more of cluster close to normal distribution.

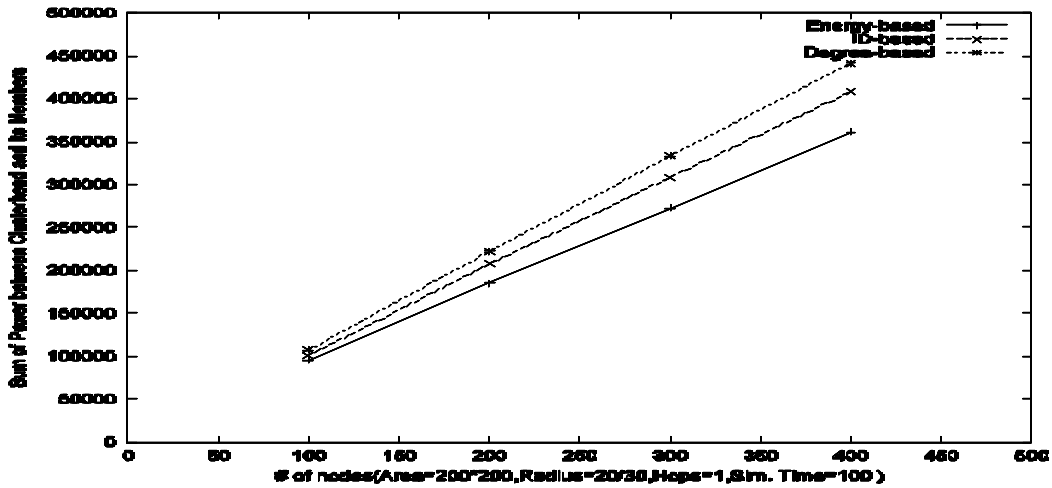


Fig 4　The energy for communication between clusterhead and its members .

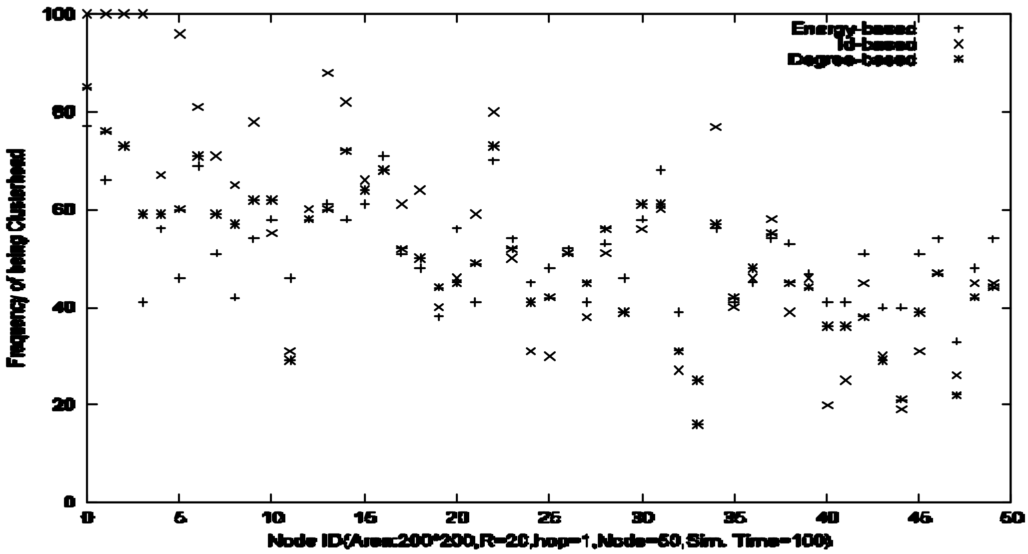


Fig 5　The status of nodes works as a clusterhead(100 times).

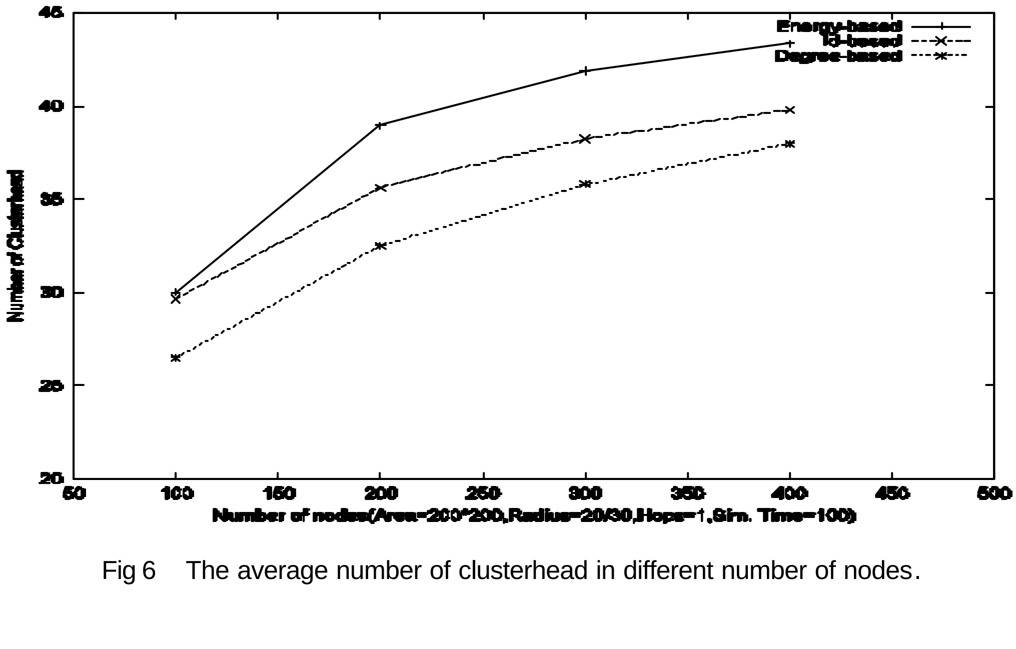


Fig 6　The average number of clusterhead in different number of nodes.

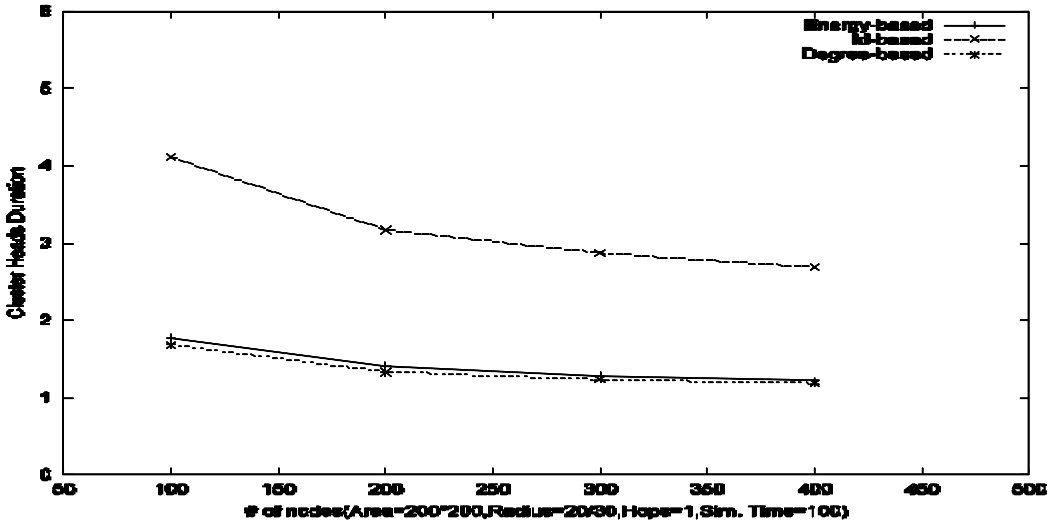


Fig 7 The clusterhead duration in different number of nodes. ECEP is closed to Degree-based.

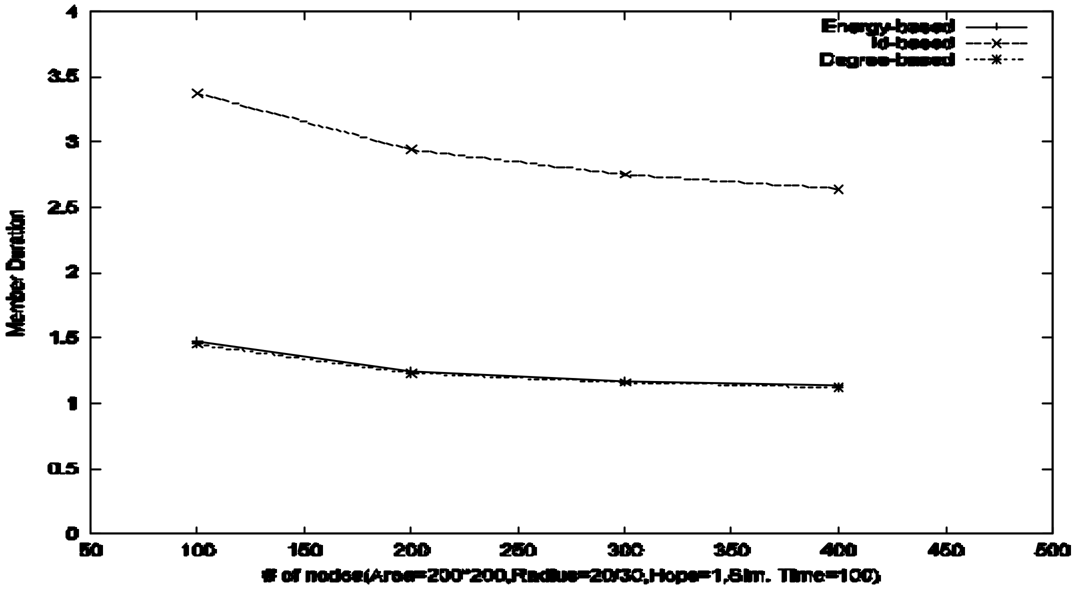


Fig 8 The member duration in different number of nodes. ECEP is closed to Degree-based.