



Revista Facultad de Ingeniería Universidad de Antioquia

ISSN: 0120-6230

revista.ingenieria@udea.edu.co

Universidad de Antioquia  
Colombia

Correa, Blanca Alicia; Ospina, Laura; Hincapié, Roberto Carlos  
Survey of clustering techniques for mobile ad hoc networks  
Revista Facultad de Ingeniería Universidad de Antioquia, núm. 41, septiembre, 2007, pp. 145-161  
Universidad de Antioquia  
Medellín, Colombia

Available in: <http://www.redalyc.org/articulo.oa?id=43004112>

- How to cite
- Complete issue
- More information about this article
- Journal's homepage in redalyc.org

redalyc.org

Scientific Information System  
Network of Scientific Journals from Latin America, the Caribbean, Spain and Portugal  
Non-profit academic project, developed under the open access initiative

## Survey of clustering techniques for mobile ad hoc networks

## Técnicas de agrupamiento para redes móviles ad hoc

*Blanca Alicia Correa\*, Laura Ospina, Roberto Carlos Hincapié*

Universidad Pontificia Bolivariana, Grupo de Investigación, Desarrollo y Aplicación en Telecomunicaciones e Informática (GIDATI). Facultad de Ingeniería. Circular 1.ª N.º 70-01. Medellín, Colombia.

(Recibido el 25 enero de 2007. Aceptado el 12 de abril de 2007)

### Abstract

Clustering methods allow fast connection and also better routing and topology management of mobile ad hoc networks (MANET). In this paper a survey of clustering techniques for MANET is presented and some preliminary concepts for designing clustering algorithms are introduced. These concepts relate to network topology, routing schemes, graph partitioning and mobility algorithms. In addition, some of the most popular clustering techniques, such as Lowest-ID heuristic, Highest degree heuristic, DMAC (distributed mobility-adaptive clustering), and WCA (weighted clustering algorithm), among other techniques are described. The aim is to illustrate the main concepts concerning clustering techniques in mobile ad hoc networks.

----- *Key words:* Mobile ad hoc networks, MANET, clustering, clusterhead.

### Resumen

Los métodos de agrupamiento permiten que las *MANET* (redes móviles ad hoc) presenten un mejor desempeño en cuanto a la rapidez de conexión, el enrutamiento y el manejo de la topología. En este trabajo se presenta una revisión sobre las técnicas de agrupamiento para *MANET*. Se introducen algunos temas preliminares que forman la base para el desarrollo de los algoritmos de agrupamiento, tales como: la topología de la red, el enrutamiento, la teoría de grafos y los algoritmos de movilidad. Adicionalmente, se describen algunas de las técnicas de agrupamiento más

---

\* Autor de correspondencia: teléfono: +57 +4 +312 69 49, correo electrónico: blancalim@yahoo.com (B. A. Correa).

conocidas como *Lowest-ID heuristic*, *Highest degree heuristic*, *DMAC (distributed mobility-adaptive clustering)*, *WCA (weighted clustering algorithm)*, entre otros. El propósito central es ilustrar los conceptos principales respecto a las técnicas de agrupamiento en *MANET*.

----- *Palabras clave:* redes móviles ad hoc, *MANET*, agrupamiento (*clustering*), *clusterhead*.

## Introduction

Nowadays, wireless technologies are becoming quite common in our daily lives. They have been gaining popularity with the use of portable devices like laptop computers, personal digital assistants (PDAs), and mobile phones. In order to use these devices some type of fixed infrastructure is normally required such as access points or base stations. This means that unless mobile users of wireless technologies have the possibility to access a static network, they will not be able to support their mobile devices services. Mobile ad hoc networks (MANET) propose a solution to these kinds of problems. MANET are autonomous systems consisting of mobile hosts that are connected by multi-hop wireless links. The main idea of a MANET is that a network can be established without the need for any centralized administration or fixed infrastructure. As has been found in the literature [1] ad hoc networks offer the vision of true ubiquitous computing, providing connectivity to everyone, anywhere and from any device.

Typical applications of MANET include rescue operations, military scenarios and cases in which it is impossible to establish a wired backbone. This communication technology can also be used for extending the coverage of current wireless networks. In future applications MANET could be deployed by students using laptop computers participating in an interactive activity or by people at a meeting using their PDAs or laptops to exchange information among themselves [2].

However, the implementation of MANET for commercial purposes is not an easy task. Unlike other wireless technologies such as cellular networks, MANET present mayor challenges to the research community because topologies in these networks are random, multihop and dynamic. In addition, link bandwidth and mobile nodes transmission power are scarce [3]. Consequently, MANET face more difficult problems concerning management functions, routing and scalability [4]. As a solution to these complications, clustering schemes are proposed for MANET in order to

organize the network topology in a hierarchical manner. Many clustering techniques have been developed. In this survey we intend to expose some of these techniques and the large benefits that come along when a clustering approach is implemented in MANET.

The main purpose of this work is to explain basic concepts concerning clustering techniques in mobile ad hoc networks to students engaging in this field of study. This paper is organized as follows. A short overview about MANET is presented in section 2. Routing schemes for MANET are explained in section 3. Next, in section 4, the concept of topology control in ad hoc networks is introduced. Concepts concerning clustering in MANET are described in section 5. Graph concepts applied to ad hoc networks are explained in section 6. A review of some known clustering techniques is presented in section 7. In Section 8 the clustering schemes explained are discussed. Finally, in section 9 we arrive to the main conclusions of this survey.

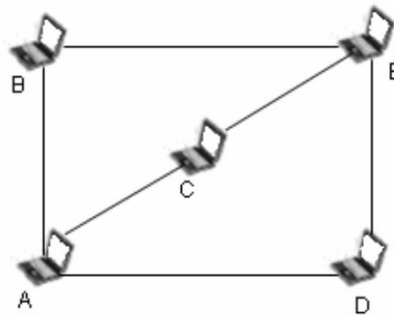
## Preliminary concepts of MANET

Wireless networks include infrastructure-based networks and ad hoc networks. Most wireless infrastructure-based networks are established by a one hop radio connection to a wired network. On the other hand, mobile ad hoc networks are decentralized networks that develop through self-organization [5]. The original idea of MANET started out in the early 1970s. At this time they were known as packet radio networks. Lately, substantial progress has been made in technologies like microelectronics, wireless signal processing, distributed computing and VLSI (Very Large Scale Integration) circuit design and manufacturing [6]. This has given the possibility to put together node and network devices in order to create wireless communications with ad hoc capability.

MANET are formed by a group of nodes that can transmit and receive data and also relay data among themselves. Communication between nodes is made over wireless links. A pair of nodes can establish a wireless link among themselves

only if they are within transmission range of each other. An important feature of ad hoc networks is that routes between two hosts may consist of hops through other hosts in the network [7]. When a sender node wants to communicate with a receiver node, it may happen that they are not within communication range of each other. However, they might have the chance to communicate if other hosts that lie in-between are willing to forward

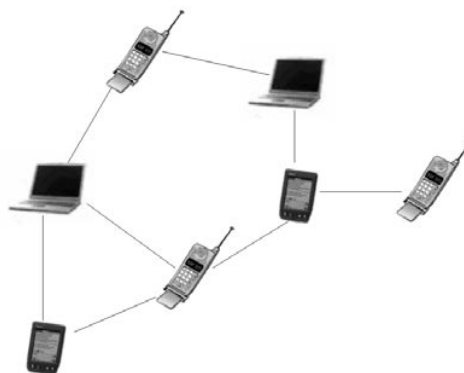
packets for them [8]. This characteristic of MANET is known as multihopping. An example is shown in figure 1. Node A can communicate directly (single-hop) with node B, node C and node D. If A wants to communicate with node E, node C must serve as an intermediate node for communication between them. Therefore, the communication between nodes A and E is multi-hop.



**Figure 1** Multi-hop communication in a mobile ad hoc network

Today wireless bluetooth, personal area networks (PAN), IEEE 802.11 a/b/g, wireless local area networks (WLAN) and HIPERLAN/2, are communication standards that include ad hoc

features [9]. Figure 2 shows an example of a mobile ad hoc network composed of commonly used wireless devices.



**Figure 2** Wireless mobile ad hoc network

### Routing schemes in MANET

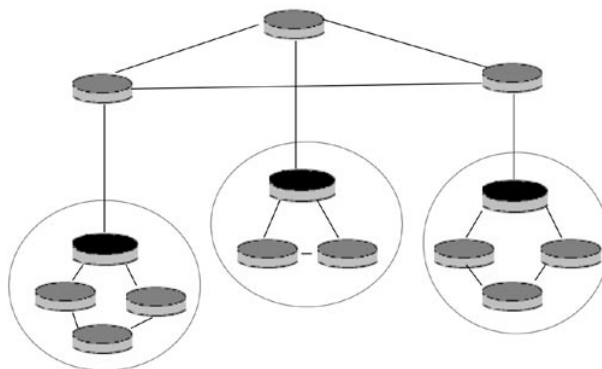
Routing in ad hoc networks is different compared to normal wired networks. The use of conventional routing protocols in a dynamic network is inconvenient because they place a heavy computational burden on mobile computers and they present convergence characteristics that don't suit well enough the needs of dynamic networks [10]. For instance, any routing scheme in a dynamic environment such as ad hoc networks must consider that the topology of the network can vary while the packet is being routed [11] and that the quality of wireless links is highly variable. In wired networks, link failure is not frequent since the network structure is mostly static. Therefore, routes in MANET must be calculated much more frequently in order to keep up the same response level of wired networks [12]. Routing schemes in MANET are classified in four major groups, namely, flooding, proactive routing, reactive routing, and hybrid routing.

Flooding is used in MANET to propagate control messages. Flooding is a distributed process [8] in which a node transmits a message to all its neighbours and these transmit the message consecutively to their neighbours and so on until the message has been disseminated to the entire network [13]. Although flooding is the simplest way to establish communication in MANET, it is inefficient [8] and generates too much traffic.

In pro-active routing (table-driven), valid routes are maintained to every node all the time [2]. Updates are propagated throughout the network when a change in the network topology occurs [14]. Proactive routing is only appropriate for small networks because as networks grow in size the overhead increases.

In reactive routing (on-demand) the route evaluation is done only when it is necessary. When a node needs to find a route to another destiny node it must begin a discovery process to find one that is appropriate. Paths are maintained only until they are needed.

Hybrid routing provides routing through the implementation of a hierarchical approach [2]. In a hierarchical approach the network is organized into subsets of nodes, known as clusters. This topology organization reduces network traffic because a node only needs to have knowledge of the routing information within its cluster and not of the entire network. Hybrid routing, also known as cluster-based routing is a convenient scheme for developing efficient routing algorithms in MANET. Apart from making a large network appear smaller, one significant feature of cluster-based routing is that it can make a dynamic topology appear less dynamic [15]. In order to implement a dynamic hybrid routing scheme, efficient clustering algorithms must be designed. In figure 3 a hierarchical network organization is shown.



**Figure 3** Hierarchical network organization

### **Topology control**

Topology control deals with the problem of computing and maintaining a connected topology among nodes in ad hoc networks [16]. Topology control covers: power control and hierarchical topology organization. Power control ensures network connectivity by adjusting the power of each node in order to balance one-hop neighbour connectivity [17]. On the other hand, hierarchical topology control is an approach often referred to as clustering [18]. For the rest of this paper we will focus on this approach.

### **Clustering in MANET**

A successful approach for dealing with the maintenance of mobile ad hoc networks is by partitioning the network into clusters. In this way the network becomes more manageable. It must be clear though that a clustering technique is not a routing protocol. Clustering is a method which aggregates nodes into groups [4]. These groups are contained by the network and they are known as clusters. A cluster is basically a subset of nodes of the network that satisfies a certain property [19]. Clusters are analogous to cells in a cellular network. However, the cluster organization of an ad hoc network cannot be achieved offline as in fixed networks [15].

Clustering presents several advantages for the medium access layer and the network layer in MANET [20]. The implementation of clustering schemes allows a better performance of the protocols for the Medium Access Control (MAC) layer by improving spatial reuse, throughput, scalability and power consumption. On the other hand, clustering helps improve routing at the network layer by reducing the size of the routing tables and by decreasing transmission overhead due to the update of routing tables after topological changes occur [7, 19]. Clustering helps aggregate topology information since the number of nodes of a cluster is smaller than the number of nodes of the entire network. Therefore, each node only needs to store a fraction of the total network routing information [21].

The purpose of a clustering algorithm is to produce and maintain a connected cluster [22]. In most clustering techniques nodes are selected to play different roles according to a certain criteria. In general, three types of nodes are defined:

#### **Ordinary nodes**

Ordinary nodes are members of a cluster which do not have neighbours belonging to a different cluster [23].

#### **Gateway nodes**

Gateway nodes are nodes in a non-clusterhead state located at the periphery of a cluster. These types of nodes are called gateways because they are able to listen to transmissions from another node which is in a different cluster [21]. To accomplish this, a gateway node must have at least one neighbour that is a member of another cluster [24].

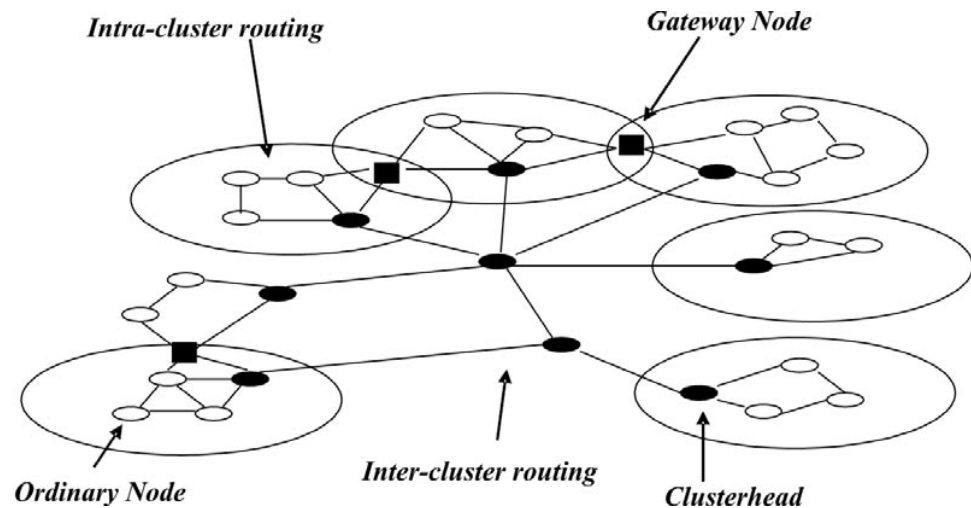
#### **Clusterheads**

Most clustering approaches for mobile ad hoc networks select a subset of nodes in order to form a network backbone that supports control functions. A set of the selected nodes are called clusterheads and each node in the network is associated with one. Clusterheads are connected with one another directly or through gateway nodes. The union of gateway nodes and clusterheads form a connected backbone. This connected backbone helps simplify functions such as channel access, bandwidth allocation, routing power control and virtual-circuit support [18].

Clusterheads are analogous to the base station concept in current cellular systems. They act as local coordinators in resolving channel scheduling and performing power control [25]. However, the difference of a clusterhead from a conventional base station resides in the fact that a clusterhead does not have special hardware, it is selected among the set of stations and it presents a dynamic and mobile behavior [26]. Since clusterheads must perform extra work with respect to ordinary

nodes they can easily become a single point of failure within a cluster. For this reason, the clusterhead election process should consider for the clusterhead role, those nodes with a higher degree of relative stability [4]. The main task of a clusterhead is to calculate the routes for long-distance messages and to forward inter-cluster packets. A packet from any source node is first directed to its clusterhead. If the destination is

located in the same cluster, the clusterhead just forwards the packet to the destination node. If the destination node is located in a different cluster, the clusterhead of the sending node routes the packet within the substructure of the network, to the clusterhead of the destination node. Then, this clusterhead forwards the packet to its final destiny [12]. Figure 4 shows different roles of nodes in a mobile ad hoc network organized by clusters.



**Figure 4** Cluster configuration of an ad hoc network and node roles

### Graph theory

A multi-hop ad hoc network is normally represented by an undirected graph  $G = (V, E)$ .  $V$  represents the set of nodes and  $E$  is the edge set that represents the set of links between nodes assuming all hosts have the same transmission range  $R$ . Two hosts are considered neighbours if and only if their geographic distance is less than  $R$  [27]. Therefore, two nodes are connected by an edge if they can communicate with each other over bi-directional links. The movement of the mobile hosts is only implicitly considered [8]. The distance between two nodes in a graph

corresponds to the number of edges that together form the shortest path between them [1].

In order to study wireless ad hoc networks, a special class of graphs known as unit disk graphs is employed in order to model these networks. Formally, unit disk graphs are defined as the intersection graphs of unit disks in the plane [28]. In general, unit disk graphs allow broadcast networks and some problems in computational geometry to be described by a graph-theoretical model [29]. Unit disks graphs model broadcast networks by representing every node by a point in the graph and by representing the effective broadcast range of each point as a unit disk [1].

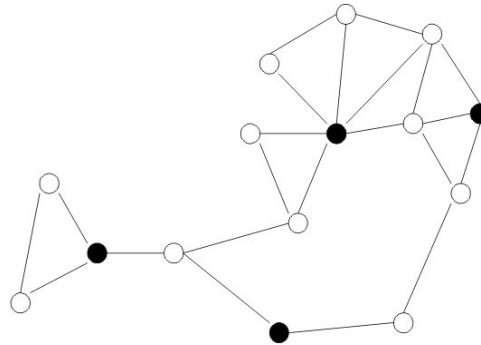


### **Dominating sets**

Most clustering techniques in ad hoc networks are based on dominating sets. Distributed algorithms are designed in order to find small dominating sets. The term distributed means that each node only has knowledge of the local structure of the network [11]. A set  $S$  is dominating if each node in the graph  $G = (V, E)$  is either in  $S$  or adjacent to at least one of the nodes in  $S$  [30]. A dominant set in mobile ad hoc networks is formed by the set of the network clusterheads. Since a large number of clusterheads will increase substantially the number of routing hops causing higher communication overheads and extra energy consumption [31], the basic idea is to find a minimum dominating set. The minimum dominating set problem is formulated in the context of a network in order to find a minimum number of transmitters that allow all the other stations to be within the range of at least one of the selected transmitters [28].

However, the computational problem of finding a minimum dominating set is NP-hard, which means that no polynomial time algorithm can guarantee an optimal solution.

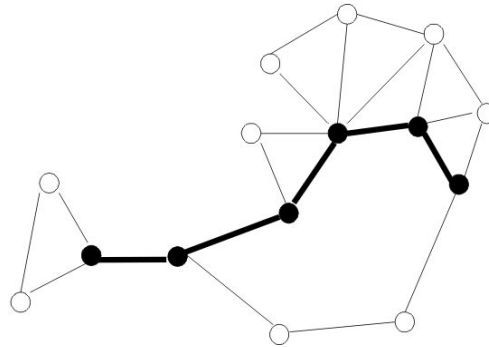
A dominating set can also be independent, connected or weakly connected. In an independent dominating set no two vertices are adjacent. Figure 5 shows an example of an independent dominating set. However, as the topology changes, the use of independent dominating sets can cause the so-called chain reaction. This happens when two clusterheads become adjacent and one of them has to abdicate. This may cause other clusterhead changes to propagate through the network. The chain reaction is a cluster maintenance problem and it can be avoided by removing the independent condition on dominating sets [6, 12]. There are several clustering algorithms based on independent dominating sets of graphs such as the Lowest ID and Highest degree heuristics [32, 33] and DMAC [34].



**Figure 5** Independent dominating set

A connected dominating set (CDS) is a dominating set in which the induced subgraph is connected. A connected dominating set favours routing but the connectivity constraint originates a relatively large number of clusters [35]. Figure 6 illustrates a connected dominating set.

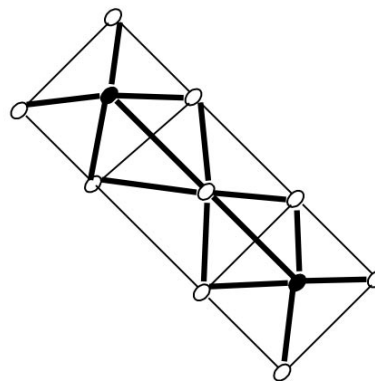
The number of clusters can be reduced by relaxing the connectivity requirement. This is achieved by a weakly connected dominating set (WCDS). A subgraph weakly induced by a dominating set  $S$  is the graph  $G' = (V, E')$  such that each edge in  $E'$  has at least one endpoint in



**Figure 6** Connected dominating set

$S$ . The set  $S$  is a weakly connected dominating set if  $S$  is dominating and  $G'$  is connected [30]. A simple way to see this is by connecting every two vertices of a dominating set that are at a distance of 1 or 2. If the induced subgraph is connected, then the dominating set is weakly

connected [36]. WCDS can be smaller than CDS and can help simplify the cluster formation since fewer clusters are created. In figure 7 the black circles are the weakly-connected dominating set and the black edges illustrate the weakly induced subgraph.



**Figure 7** Weakly-connected dominating set

### Clustering Algorithms in MANET

In this section we describe eight clustering algorithms known as: Lowest ID heuristic [32], Highest degree heuristic [33], k-CONID [27], Max-min heuristic [39],  $(\alpha, t)$  cluster framework

[15], MobDhop [21], DMAC [34] and WCA [40]. These schemes will be discussed more deeply although other interesting approaches include the clustering algorithms presented in [37] and [38] which make use of fuzzy set theory to form

dynamic clusters. In [37] nodes are considered data objects characterized by certain attributes and the membership of the nodes to the clusters is treated in a fuzzy way.

There are several types of clustering schemes. Some are based on the selection of a clusterhead [21, 27, 32, 33, 34, 39, 40], and others eliminate this requirement [15, 26]. Although a clusterhead can dictate consensus, it may easily become the traffic bottleneck and single point of failure of the cluster [26]. In order to avoid this, an algorithm has been proposed in which the requirement for a clusterhead is eliminated and a fully distributed approach for cluster formation and intracluster communications is adopted [26]. The objective of this clustering algorithm is to find an interconnected set of clusters covering the entire node population.

There are clustering schemes based on multi-hop clusters [21, 27, 39] and others that only form 1-hop clusters [32, 33, 34, 40].

#### ***Lowest-ID heuristic and Highest degree heuristic***

The Lowest-ID and the Highest degree algorithms are two of the most popular clustering algorithms proposed for ad hoc networks. Many recent clustering techniques have used these two algorithms as a reference to compare their heuristics performance.

These two weight-based clustering algorithms are proposed for networking purposes such as multimedia traffic support [25]. The Lowest-ID and Highest-degree algorithms make use of node ID and node degree for determining weights respectively. Both algorithms are localized since a node only needs to know the information of its 1-hop neighbors.

In the Lowest-ID algorithm, every node in the network has a designated unique identifier (ID). Nodes periodically broadcast their ID to their direct neighbors. Each node compares the IDs of its neighbors with its own ID. A node decides

to become a clusterhead if it has the lowest ID among its neighbors IDs.

The highest degree heuristic is based on the connectivity between a node and its direct neighbors. Each node broadcasts periodically its connectivity value to its direct neighbors. A node decides to become a clusterhead or remain an ordinary node by comparing the connectivity value of its neighbors with its own value. If a node has the highest connectivity value in its neighborhood it will become a clusterhead.

#### ***$(\alpha, t)$ cluster framework***

The  $(\alpha, t)$  clustering strategy [15, 41, 42] is a simple distributed asynchronous algorithm that uses a probability model for determining path availability in order to make clustering decisions. This algorithm organizes nodes into clusters in a dynamic way. In the  $(\alpha, t)$  approach it is attempted to provide an effective topology that adapts to node mobility.

Path availability is a random process which is determined by the mobility of nodes that lie along a certain path. In the  $(\alpha, t)$  approach paths are evaluated by two system parameters,  $\alpha$  and  $t$ .  $\alpha$  establishes a lower bound on the probability that a given cluster path will remain available for a time  $t$ .  $\alpha$  controls cluster stability while the role of  $t$  is to manage cluster size for a given level of stability.

The actions taken by the clustering algorithm depend upon the information given by the routing and network-interface layer protocol. Each node in the network is given a node's cluster identifier number (CID) and makes use of a timer named  $\alpha$  timer. This timer establishes the maximum time  $t$  for which a node guarantees that paths will be available to each cluster destination with probability  $\geq \alpha$ .

In the  $(\alpha, t)$  algorithm, clusters which satisfy the  $(\alpha, t)$  criteria are maintained. The  $(\alpha, t)$  criteria is accomplished if the probabilistic bound  $\alpha$  on the mutual availability of paths between nodes in

a cluster exists over a specified interval of time  $t$ . Therefore, the algorithm applies prediction of node mobility as criteria for cluster organization. The  $(\alpha, t)$  algorithm characterizes the probability of link and path availability as a function of a random walk mobility model.

The algorithm is designed to take appropriate actions upon topological changes. A topological change requires that nodes reevaluate the  $(\alpha, t)$  criteria. The documentation that supports this clustering approach presents the pseudocode for five important topological changes that determine the  $(\alpha, t)$  cluster algorithm: Node activation, link activation, link failure, node deactivation and  $\alpha$  timer expiration.

#### ***k-CONID***

This approach combines two clustering algorithms: the Lowest-ID and the Highest-degree heuristics. The clustering algorithm [27] is named  $k$ -CONID ( $k$ -hop connectivity ID). In order to select clusterheads connectivity is considered as a first criterion and lower ID as a secondary criterion. Using only node connectivity as a criterion causes numerous ties between nodes [27]. On the other hand, using only a lower ID criterion generates more clusters than necessary. The purpose is to minimize the number of clusters formed in the network and in this way obtain dominating sets of smaller sizes. Clusters in the  $k$ -CONID approach are formed by a clusterhead and all nodes that are at distance at most  $k$ -hops from the clusterhead.

At the beginning of the algorithm, a node starts a flooding process in which a clustering request is sent to all other nodes. In the Highest-degree heuristic, node degree only measures connectivity for 1-hop clusters.  $k$ -CONID generalizes connectivity for a  $k$ -hop neighbourhood. Thus, when  $k = 1$  connectivity is the same as node degree.

Each node in the network is assigned a pair  $did = (d, ID)$ .  $d$  is a node's connectivity and  $ID$  is the node's identifier. A node is selected as a clusterhead if it has the highest connectivity. In

case of equal connectivity, a node has clusterhead priority if it has lowest ID. The basic idea is that every node broadcasts its clustering decision once all its  $k$ -hop neighbours with larger clusterhead priority have done so.

Although each node only determines one cluster, clusters may overlap. This means that a node can belong to all clusters whose clusterhead is at most  $k$ -hops distance from the node. Nodes that belong to more than one cluster become gateway nodes.

#### ***Max-min d-cluster***

In many papers the clusterhead election is done in such a way that no node can be more than one hop away from its clusterhead. The main drawback of nodes being 1-hop away from the clusterhead, is the generation of a large number of clusterheads within the network causing a congestion problem [39]. Therefore, in the Max-Min heuristic clusters are formed by nodes that can be at most  $d$ -hops away from the clusterhead. A  $d$ -neighbourhood of a node consists of the node itself and the set of all nodes located within  $d$ -hops away from the node.

In this heuristic  $d$  is defined as the maximum number of hops away from the nearest clusterhead ( $d \geq 1$ ). This value is an input to the heuristic allowing control over the number of clusterheads to be selected. Nodes participate in the clusterhead election algorithm based on their node id. Each node has two arrays named WINNER and SENDER. The first array corresponds to the winner node id of a particular round. The latter array refers to the node that sent the winning node id for a particular round. Once the clusterhead is selected, SENDER helps determine the shortest path back to the clusterhead.

At the beginning of the algorithm every node has a WINNER value equal to its own node id. The algorithm initiates  $2d$  rounds of flooding. In this way nodes exchange their node ids. The 1<sup>st</sup>  $d$ -round executes a procedure named Floodmax and the 2<sup>nd</sup>  $d$ -round executes a procedure named

Floodmin. Every node in the network records the WINNER value for each flooding round. The procedure in each  $d$ -round is as follows:

- Floodmax: Each node broadcasts its WINNER value to all its 1-hop neighbours. For every  $d$ -round of Floodmax a node determines its new WINNER value as the largest value among its received WINNER values and its own value.
- Floodmin: This procedure follows Floodmax and it begins with the values that exist at each node after the 1<sup>st</sup>  $d$ -rounds of Floodmax. In this case in each  $d$ -round, a node chooses for its new WINNER value the smallest value among its received WINNER values and its own.

The clusterhead selection criterion is based on the registered entries of each node after the  $2d$  rounds of flooding. A node can declare itself as a clusterhead if it has received its original node id after the  $2d$  rounds of flooding. If not each node should look for node pairs. A node pair is defined as any node id that results as a WINNER at least once in the 1<sup>st</sup>  $d$ -rounds of Floodmax as well as in the 2<sup>nd</sup>  $d$ -rounds of Floodmin for an individual node. After a node has detected all node pairs it chooses the minimum node pair as clusterhead. Therefore, the smallest node id appearing in both of the flooding stages is chosen as the clusterhead. If a node does not identify any node pair it selects the maximum node id in the 1<sup>st</sup>  $d$ -rounds of flooding as its clusterhead.

Finally, each node broadcasts its selected clusterhead to all its neighbors. A node determines it is a gateway node if it has neighbouring nodes with different clusterheads. After a node has identified that it is a gateway node it sends topological information to its clusterhead. The SENDER array is used to decide who next to send this information to. In this manner a clusterhead can obtain information of every node in its cluster. This heuristic maximizes the number of gateways in order to obtain a backbone formed by multiple paths between neighbouring clusterheads. In this way, congestion can be eased.

### **MobDhop**

A clustering algorithm has been reported which partitions an *ad hoc* network into  $d$ -hop clusters based on a mobility metric [21]. The distributed algorithm is called MobDhop. The objective of forming  $d$ -hop clusters is to make the cluster diameter more flexible. MobDhop is based on mobility metric [43] and the diameter of a cluster is adaptable with respect to node mobility. This clustering algorithm assumes that each node can measure its received signal strength. In this manner, a node can determine the closeness of its neighbours. Strong received signal strength implies closeness between two nodes. The MobDhop algorithm requires the calculation of five terms: the estimated distance between nodes, the relative mobility between nodes, the variation of estimated distance over time, the local stability, and the estimated mean distance. A node calculates its estimated distance to a neighbour based on the measured received signal strength from that neighbour. Relative mobility corresponds to the difference of the estimated distance of one node with respect to another, at two successive time moments. This parameter indicates if two nodes move away from each other or if they become closer.

The variation of estimated distances between two nodes is computed instead of calculating physical distance between two nodes. This is because physical distance between two nodes is not a precise measure of closeness. For instance, if a node runs out of energy it will transmit packets at low power acting as a distanced node from its physically close neighbour. The variation of estimated distance and the relative mobility between nodes are used to calculate the local stability. Local stability is computed in order to select some nodes as clusterheads. A node may become a clusterhead if it is found to be the most stable node among its neighbourhood. Thus, the clusterhead will be the node with the lowest value of local stability among its neighbours.

MobDhop is executed in three stages as follows:

- *Discovery stage:* Initially, nonoverlapping clusters formed by a clusterhead and its one-hop neighbors are created. After a discovery period in which nodes acquire complete knowledge of their neighbourhoods, each node computes its local stability value in order to select some nodes as clusterheads. If a node can hear messages from a node that belongs to a different cluster, it becomes a gateway node. If not, it becomes a cluster member.
- *Merging stage:* The two-hop clusters established in the discovery stage are expanded by a merging process. A merging process can be initiated by a nonclustered node that requests to join its neighbouring clusters or when two neighbouring gateways request to merge their clusters.
- *Cluster maintenance stage:* A cluster maintenance stage is invoked when topology changes occur. A topology change can happen when a node switches on. This node will begin the merging process in order to join a cluster. Another case that causes a topology change is when a clusterhead switches off. In this case, its immediate neighbours begin the discovery process so that a new clusterhead can be selected.

### DMAC

DMAC is a distributed algorithm in which clusterheads are selected using a weight-based criterion that depends on node mobility-related parameters. This algorithm is suited to manage highly mobile networks. DMAC overcomes a mayor drawback found in most clustering algorithms. A common assumption that is presented in most algorithms is that during the set up time nodes do not move while they are being grouped into clusters. Normally, clustering algorithms partition the network into clusters and only after this step has been accomplished, the non mobility assumption is released. Afterwards, the algorithm tries to maintain the cluster topology as nodes move. In real ad hoc

situations this assumption can not be made due to the constant mobility of nodes. Therefore one important feature of DMAC is that nodes can move, even during the clustering set up.

During the algorithm execution it is assumed that each node has a weight (a real number  $\geq 0$ ) and an ID (node's unique identifier) associated to it. The weight of a node represents node mobility parameters. A node chooses its own role (clusterhead or ordinary node) based on the knowledge of its current one hop neighbors. A node becomes a clusterhead if it has the highest weight among its one-hop neighbours; otherwise it joins a neighbouring clusterhead.

During execution of this algorithm, every node has knowledge of its ID, its weight as well as its neighbors ID and its neighbor's weight. DMAC is a message driven algorithm (except during the initial procedure). Two types of messages are used: If a node joins a cluster it sends out a Join message and if it becomes a clusterhead it sends a CH message. A node decides its own role once all its neighbors with bigger weights have decided their roles.

DMAC executes five procedures at each node: an init routine, a link failure procedure, a new link procedure, a procedure depending upon the reception of a CH message and a procedure depending upon the reception of a Join message. When a clusterhead receives a Join message from an ordinary node, it checks if the sending node is joining its own cluster or a different one. On the other hand, if an ordinary node receives a Join message from its clusterhead, it means that this clusterhead is giving up his role. Upon the reception of a CH message a node checks if it will affiliate or not to the sending clusterhead.

The adaptation feature of the clustering algorithm is made possible by letting each node react to the failure of a link with another node or to the presence of a new link. Upon link failure between a clusterhead and one of his node members, the membership of the node to the cluster is removed, and this node must determine its new role. A new

link between two nodes means that a node has detected the presence of a new neighbour. In this case, the node must determine if this new node has a larger weight than its own clusterhead in order to join it. If the node is a clusterhead itself then it will give up its role if the new clusterhead has a higher weight.

### WCA

A distributed clustering algorithm based on weight values has been proposed [40]. The weighted clustering algorithm (WCA) selects clusterheads by considering important aspects related to the efficient functioning of the system components. Therefore, in order to optimize battery usage, load balancing and MAC functionality a node is chosen to be a clusterhead according to the number of nodes it can handle, mobility, transmission power and battery power. To avoid communications overhead, this algorithm is not periodic and the clusterhead election procedure is only invoked based on node mobility and when the current dominant set is incapable to cover all the nodes. To ensure that clusterheads will not be over-loaded a pre-defined threshold is established in order to specify the number of nodes each clusterhead can ideally support. This parameter corresponds to  $\delta$ . WCA selects the clusterheads according to the weight value of each node. The weight associated to a node  $v$  is defined as:

$$W_v = w_1\Delta_v + w_2D_v + w_3M_v + w_4P_v$$

The node with the minimum weight is selected as a clusterhead. The weighting factors are chosen so that  $w_1 + w_2 + w_3 + w_4 = 1$ .  $M_v$  is the measure of mobility. It is taken by computing the running average speed of every node during a specified time  $T$ .  $\Delta_v$  is the degree difference.  $\Delta_v$  is obtained by first calculating the number of neighbours of each node. The result of this calculation is defined as the degree of a node  $v$ ,  $dv$ . To ensure load balancing the degree difference  $\Delta_v$  is calculated as  $|dv - \delta|$  for every node  $v$ . The parameter  $D_v$  is defined as the sum of distances from a given node to all its neighbours. This factor is related to energy consumption since more power is

needed for larger distance communications. The parameter  $P_v$  is the cumulative time of a node being a clusterhead.  $P_v$  is a measure of how much battery power has been consumed. A clusterhead consumes more battery than an ordinary node because it has extra responsibilities.

The clusterhead election algorithm finishes once all the nodes become either a clusterhead or a member of a clusterhead. The distance between members of a clusterhead, must be less or equal to the transmission range between them. No two clusterheads can be immediate neighbours.

There has also been interesting work presented in which WCA is optimized by advanced computational methods such as genetic algorithms [44], simulated annealing [45] and particle swarm optimization [46]. Clustering schemes are summarized in table 1.

### Discussion

The clustering schemes presented still face several difficulties that must be overcome. For instance, some drawbacks presented by the Lowest-ID and the Highest-degree algorithm have to do with the configuration of unstable clusters. In the case of the Lowest-ID, a highly mobile node with the lowest ID among its neighbours can be selected as a clusterhead causing inconvenient re-clustering and undesired clusterhead changes in the network. However, this could be minimized by selecting a clusterhead that stays in a particular cluster for a long time and keeps the cluster-size nearly constant [3].

In the  $(\alpha, t)$  cluster framework a link is considered to be active between two mobile nodes at time  $t_l + t_0$  ( $t_l > t_0$ ) if at time  $t_0$  there is an active link between them. This clustering algorithm leads to an ambiguity as to how big  $t_l$  is and it does not take into consideration events that might have taken place in the interval  $t_l + t_0$  [3].

The max-min  $d$ -cluster scheme is based on flooding which causes network overhead because too many messages have to be sent. The implementation of  $d$ -hop clusters [39] does not take into account

**Table1** Summary of clustering schemes for mobile ad hoc networks

<i>Clustering Scheme</i>	<i>Characteristics</i>	<i>Clusterhead selection criteria</i>
Lowest-ID [32]	Based on node ID	Lowest node ID among one-hop neighbors
Highest degree [33]	Based on highest degree (connectivity value)	Highest connectivity value among its direct neighbors
k-CONID [27]	Combines Highest-degree and Lowest ID Forms k-hop clusters	Highest degree as a first criterion and node ID as a second criterion
$(\alpha, t)$ cluster framework [39]	Clusters are formed based on path availability	No clusterheads are selected
Max-min d-cluster [15]	d-hop clusters are formed Uses node id Cluster formation is based on $2d$ rounds of flooding	Three rules based on the registered entries of each node after $2d$ rounds of flooding are followed
MobDhop [21]	d-hop clusters are formed Based on the received signal strength by each node	Lowest value of local stability among neighbors
DMAC [34]	Cluster formation is mostly message-driven Each node is assigned a weight value and an ID	Highest weight value among one-hop neighbours
WCA [40]	A combined weight metric based on the number of nodes handled, mobility, transmission power and battery power is defined	Minimum weight value among neighbours

node mobility and the node mobility pattern [21]. If node mobility pattern is considered, maximum cluster stability can be ensured [21].

A mayor drawback presented by DMAC is that as node density increases, clusterheads may become overloaded. The number of node members of a cluster should be limited and existing clusters should be split into several smaller clusters, in order to solve this problem [47]. It has been shown that DMAC can trigger undesirable global rippling effects [19]. A solution to this problem is found in reference [48] where a clusterhead change only takes place when two clusterheads become 1-hop neighbors. [48].

Although WCA presents a better performance compared to other previous clustering techniques [49], there is a high reaffiliation frequency when nodes move very fast. Therefore, entropy based clustering schemes have been proposed in order to improve network stability [49, 50].

## Summary and Conclusions

In this survey we have explained concepts commonly found in many studies regarding clustering techniques for mobile ad hoc networks. We have reviewed several clustering algorithms which help organize mobile ad hoc networks in a hierarchical manner and presented some of their



main characteristics. Clustering methods improve network scalability, routing and topology management of MANET. Future work includes improving clustering schemes metrics such as node reaffiliation frequency, cluster size, stability and number of clusterheads selected.

## References

1. Y. Fernandess, D. Malkhi. "K-clustering in wireless ad hoc networks". *Proceedings of the second ACM international workshop on Principles of Mobile Computing*. 2002. pp. 31-37.
2. J.P. Grady, A. McDonald. "State of the Art: Ad Hoc Networking". *M-Zones State of the Art Paper, SOA paper 05/03*. 2003.
3. S. Sivavakeesar, G. Pavlou. "A Prediction-Based Clustering Algorithm to achieve Quality of Service in Multihop Ad Hoc Networks". *Proc. of the London Communications Symposium (LCS)*. 2002. pp. 157-160.
4. B. An, S. Papavassiliou. "A mobility-based clustering approach to support mobility management and multicast routing in mobile ad-hoc wireless networks". *International Journal of Network Management*. Vol. 11. 2001. pp. 387-395.
5. C. Prehofer, C. Bettstetter. "Self organization in communication networks: Principles and design paradigms". *IEEE Communications Magazine*. Vol. 43. Issue 7. 2005. pp. 78-85.
6. Y. P. Chen, A. L. Liestman, J. Liu. *Ad Hoc and Sensor Networks, Wireless Networks and Mobile Computing. Clustering Algorithms for Ad hoc Wireless Networks*. Vol. 2. Chapter 7: Nova Science Publishers, Hauppauge NY, 2004. pp. 145-164.
7. J. Wu, J. Cao. "Connected k-hop clustering in ad hoc networks". *ICPP*. 2005. pp 373-380.
8. I. Chatzigiannakis, S. Nikolettseas. "Design and analysis of an efficient communication strategy for hierarchical and highly changing ad-hoc mobile networks". *Mobile Networks and Applications*. Vol. 9. 2004. pp. 319-332.
9. M. Frodigh, P. Johansson, P. Larsson. "Wireless Ad Hoc Networking--The Art of Networking without a Network". *Ericsson Review*. Vol. 77. 2000. pp. 248-263.
10. P. Krishna, N.H. Vaidya, M. Chatterjee, D.K. Pradhan. "A cluster-based approach for routing in dynamic networks". *ACM SIGCOMM Computer Communication Review*. 1997. pp. 49-65.
11. S. Srivastava, R. K. Ghosh. "Cluster based routing using a k-tree core backbone for mobile ad hoc networks". *Proceedings of the 6th international workshop on discrete algorithms and methods for mobile computing and communications*. 2002. pp. 14-23.
12. Y.P. Chen, A. L. Liestman. "A zonal algorithm for clustering ad hoc networks". *International Journal of Foundations of Computer Science*. Vol.14. 2003. pp. 305-322.
13. Y. Yi, M. Gerla, T. J. Kwon. "Efficient flooding in ad hoc networks: a comparative performance study". *Proc. of the IEEE International Conference on Communications (ICC)*. Vol. 2. 2003. pp. 1059-1063.
14. S. Sesay, Z. Yang, J. He. "A survey on mobile ad hoc wireless network". *Information Technology Journal*. Vol. 3. 2004. pp. 168-175.
15. A. B. McDonald, T. F. Znati. "A mobility-based framework for adaptive clustering in wireless Ad Hoc networks". *IEEE Journal on Selected Areas in Communications*. Vol. 17. 1999. pp. 1466-1487.
16. R. Rajaraman. "Topology control and routing in ad hoc networks: a survey". *SIGACT News*, Vol. 33. 2002. pp. 60-73.
17. V. Kawadia, P. R. Kumar. "Power control and clustering in ad hoc networks". *INFOCOM 2003, Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies*. Vol.1. 2003. pp. 459-469.
18. L. Bao, J.J. Garcia-Luna-Aceves. "Topology management in ad hoc networks". *Proceedings of the 4th ACM international symposium on mobile ad hoc networking & computing*. Vol. 9. 2003. pp. 129-140.
19. L. Wang and S. Olariu. "Cluster maintenance in mobile ad-hoc networks". *Cluster Computing*. Vol. 8. 2005. pp. 111-118.
20. T. C. Hou, T. J. Tsai. "An Access-Based Clustering Protocol for Multihop Wireless Ad Hoc Networks". *IEEE J. Selected Areas Comm*. Vol. 19. 2001. pp. 1201-1210.
21. I. Er and W. Seah. "Mobility-based d-hop clustering algorithm for mobile ad hoc networks". *IEEE Wireless Communications and Networking Conference*. Vol. 4. 2004. pp. 2359-2364.
22. G. Lugano. "Clustering for hierarchical routing". 2003. [Online]. Available: <http://www.cs.helsinki.fi/u/floreen/adhoc/lugano.pdf>. Consultado julio de 2006.
23. R. Purtoosi, H. Taheri, A. Mohammadi, F. Foroozan. "A light-weight contention-based clustering algorithm for wireless ad hoc networks," *Computer and Information Technology (CIT '04)*. 2004. pp. 627-632.
24. W. Lou, J. Wu. "An enhanced message exchange mechanism in cluster-based mobile ad hoc networks". *ISPA*. 2004. pp. 223-232.

25. M. Gerla, J. Tsai. "Multicluster, mobile, multimedia radio network". *Wireless networks*. Vol. 1. 1995. pp. 255-265.
26. C.R. Lin, M. Gerla. "Adaptive clustering for mobile wireless networks". *IEEE Journal of Selected Areas in Communications*. Vol. 15. 1997. pp. 1265-1275.
27. G. Chen, F. Nocetti, J. Gonzalez, and I. Stojmenovic, "Connectivity based k-hop clustering in wireless networks". *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*. Vol. 7. 2002. pp. 188.3.
28. M. Marathe, H. Breu, H. Ravi, and D. Rosenkrantz. "Simple heuristics for unit disk graphs". 1995. [Online]. Available: <http://citeseer.ist.psu.edu/marathe95simple.html>. Consultado Julio de 2006.
29. B. Clark, C. J. Colbourn, and D. S. Johnson. "Unit disk graphs". *Discrete Mathematics*. Vol. 86. 1990. pp. 165-177.
30. K. M. Alzoubi, P.-J. Wan, and O. Frieder. "Weakly-connected dominating sets and sparse spanners in wireless ad hoc networks". *Proceedings of the 23rd International Conference on Distributed Computing Systems*. 2003. pp. 96-104.
31. J.N. Al-Karaki, A.E. Kamal, R. Ul-Mustafa. "On the optimal clustering in mobile adhoc networks". *First IEEE Consumer Communications and Networking Conference (CCNC)*. 2004. pp. 71-76.
32. A. Ephremides, J.E. Wieselthier, D.J. Baker. "A design concept for reliable mobile radio networks with frequency hopping signaling". *Proc. IEEE* 75. 1987. pp. 56-73.
33. A. Parekh. "Selecting routers in ad hoc wireless networks". *Proceedings of the SBT/IEEE International Telecommunications Symposium*. 1994.
34. S. Basagni. "Distributed clustering for ad hoc networks". *Proc. ISPAN'99 Int. Symp. On Parallel Architectures, Algorithms, and Networks*. 1999. pp. 310-315.
35. Y. P. Chen, A. L. Liestman. "Approximating minimum size weakly-connected dominating sets for clustering mobile ad hoc networks". *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing*. 2002. pp. 165-172.
36. D. Dubhashi, A. Mei, A. Panconesi, J. Radhakrishnan, A. Srinivasan. "Fast Distributed Algorithms for (Weakly) Connected Dominating Sets and Linear-Size Skeletons". *Proceedings of the Fourteenth ACM-SIAM Symposium on Discrete Algorithms, (SODA)*. 2003. pp. 717-724.
37. J. Habetha, B. Walke. "Fuzzy rule-based mobility and load management for self-organizing wireless networks". *International Journal of Wireless Information Networks*. Vol. 9. 2002. pp. 119-140.
38. I., Hwang, C. Liu, C. Wang. "Link stability based clustering and routing in ad-hoc wireless networks using fuzzy set theory". *International Journal of Wireless Information Networks*. Vol. 9. 2002. pp. 201-212.
39. A.D. Amis, R. Prakash, T.H.P. Vuong, D.T. Huynh. "Max-Min D-Cluster Formation in Wireless Ad Hoc Networks". *Proceedings of IEEE Conference on Computer Communications (INFOCOM)*. Vol. 1. 2000. pp. 32-41.
40. M. Chatterjee, S. K. Das, D. Turgut. "WCA: A Weighted clustering algorithm for mobile ad hoc networks". *Cluster Computing*. Vol. 5. 2002. pp. 193-204.
41. A. B. McDonald, T. F. Znati. "Design and Simulation of a Distributed Dynamic Clustering Algorithm for Multimode Routing in Wireless Ad Hoc Networks". *SIMULATION*. Vol. 78. 2002. pp. 408-422.
42. A. Gopalan, S. Dwivedi, T. Znati, B. McDonald. "On the Implementation and Performance of the ( $\alpha$ , t) Protocol on Linux". *SIMULATION*. Vol. 81. 2005. pp. 413-424.
43. P. Basu, N. Khan, T. D. C. Little. "A mobility based metric for clustering in mobile ad hoc networks". *Proceedings of the 21st International Conference on Distributed Computing Systems*. 2001. pp. 413.
44. D. Turgut, S.K. Das, R. Elmasri, B. Turgut. "Optimizing clustering algorithm in mobile ad hoc networks using genetic algorithmic approach". *Global Telecommunications Conference, IEEE*. Vol. 1. 2002. pp. 62-66.
45. D. Turgut, B. Turgut, R. Elmasri, T. V. Le. "Optimizing clustering algorithm in mobile ad hoc networks using simulated annealing". *Wireless Communications and Networking, IEEE*. Vol. 3. 2003. pp. 1492-1497.
46. C. Ji, Y. Zhang, S. Gao, P. Yuan, Z. Li. "Particle swarm optimization for mobile ad hoc networks clustering". *IEEE International Conference on Networking, Sensing and Control*. Vol. 1. 2004. pp. 372-375.
47. C. Bettstetter. "The Cluster Density of a Distributed Clustering Algorithm in Ad Hoc Networks". *Proc. IEEE Intern. Conf. on Communications (ICC)*. Vol. 7. 2004. pp. 4336-4340.
48. C.-C. Chiang, H.-K. Wu, W. Liu, M. Gerla. "Routing in clustered multihop, mobile wireless networks with fading channel". *IEEE Singapore International Conference on Networks (SICON)*. 1997. pp. 197-211.
49. Y. -X. Wang, F. S. Bao. "An Entropy-based Weighted Clustering Algorithm and Its Optimization for Ad Hoc Networks". 2006. [Online]. Available: [http://forrest.bao.googlepages.com/WCA\\_ETTS.pdf](http://forrest.bao.googlepages.com/WCA_ETTS.pdf). Consultado diciembre de 2006.
50. K. Robinson, D. Turgut, M. Chatterjee. "An entropy-based clustering in mobile ad hoc networks". *Proceedings of the 2006 IEEE International Conference on Networking, Sensing and Control (ICNSC)*. 2006. pp. 1-5.