Scientific Research and Essays Vol. 7(2), pp. 114-127, 16 January, 2012
Available online at http://www.academicjournals.org/SRE
DOI: 10.5897/SRE10.1102
ISSN 1992-2248 ©2012 Academic Journals

Full Length Research Paper

Fuzzy controlled multi-hop adaptive clustering (FMAC) for mobile ad hoc networks

Anuradha Banerjee[1]* and Paramartha Dutta[2]

[1]Kalyani Government Engineering College, Nadia, West Bengal, India.
[2]Visva-Bharati University, Santiniketan, West Bengal, India.

Accepted 29 August, 2011

To mimic the operations in fixed infrastructures and to solve the routing scalability problem in large 'mobile ad hoc networks', forming clusters of nodes has been proven to be a promising approach. However, when existing weighted clustering algorithms calculate each node's weight, they either consider only one metric or rely on some metrics collected from extra devices. This often leads to a higher rate of re-clustering. This article presents a robust weighted clustering algorithm called FMAC (fuzzy controlled multi-hop adaptive clustering), to form and maintain more stable clusters. In FMAC, the clusterheads are elected based on energy of nodes, rate of energy depletion, relative mobility with its neighbours, radio-range and cardinality of the set of neighbours. Two fuzzy controllers are embedded in each node for selecting clusterheads and monitoring join of nodes to clusters. Simulation results show that FMAC prolongs lifetime of ad hoc networks and has a lower clusterhead change rate and re-affiliation rate than other existing algorithms.

Key words: Adaptive, ad hoc networks, clustering, fuzzy controller, protocol.

INTRODUCTION

A mobile ad hoc network is a collection of battery-powered mobile nodes connected by relatively lower bandwidth wireless links. Each node has an area of influence called cell, only within which others can receive its transmissions. Due to no fixed infrastructures, all nodes can move freely, the network topology may change rapidly and unpredictably over time and nodes have to form their own cooperative infrastructures. Thus, each node operates as an autonomous end system and a router for others in the network. An ad hoc network is of interest because there is no prior investment for fixed infrastructures, it can be easily deployed in a short time, and end users can access and manipulate data anytime and anywhere. Examples of its applications include law enforcement operations, automated battlefield applications, natural disaster recovery situations where the communication infrastructures have been destroyed, self-organizing sensor networks for data collecting, interactive lectures or conferences for data exchanging without pre-installed infrastructures. However, these applications cannot be realized without efficient routing protocols. Scalability in ad hoc networks improves if it is divided into clusters first, and then a routing protocol is developed on top of the clustered network (Agarwal and Motwani, 2009; Correa et al., 2007; Jane et al., 2005; Yang and Zhang, 2007). A clustered ad hoc network consists of clusterheads and cluster members, where a clusterhead manages its clusters, coordinates intra/inter-cluster communication and so on. A cluster member is a node that belongs to a cluster and is not a cluster head. Many clustering algorithms have been proposed to elect clusterheads form clusters and maintain clusters (Agarwal and Motwani, 2009; Correa et al., 2007; Jane et al., 2005; Yang and Zhang, 2007). Among them we mention here some state-of-the-art protocols.

In mobility-based clustering or MOBIC (Basu et al., 2001), in order to form stable clusters, the relative mobility (RM) metric is introduced and calculated as the logarithm of ratio of received signal strengths (rss): 10 $\log_{10}$(rss1/rss2) where rss1 and rss2 are read from rss indicator when two successive 'hello' messages sent by the same neighbour are received. For each node, the

*Corresponding author. E-mail: anuradha79bn@gmail.com.

variance of RMs among its neighbours with respect to 0 is calculated as the aggregate local mobility metric. The nodes with lowest aggregate local mobility among their neighbours are elected as clusterheads. Unfortunately, it is possible that some elected clusterheads may almost run out of power, thus the re-election has to be invoked soon. Distributed clustering algorithm (DCA) (Basagni et al., 2006) elects the node that has highest node degree among its 1-hop neighbours as the clusterhead. It is suitable for networks in which nodes are static or moving at a very slow speed. DMAC (distributed modified algorithm for clustering) (Basagni et al., 2006) modifies DCA to allow node mobility during or after the cluster setup phase. In leader clustering algorithm (LCA) (www.antd.nist.gov/wahn_goals.shtml), a node becomes the clusterhead if it has the highest identification number or id, among all nodes within one hop of itself or among all nodes within one hop of one of its neighbours. LCA has a definite bias towards higher id nodes while electing clusterheads. A pathological case exists for LCA where a group of nodes are aligned in monotonically increasing order. In this case, all the nodes in the ordered sequence will become a clusterhead, generating a large number of clusterheads. LCA heuristic has been modified in LCA2 (www.antd.nist.gov/wahn_goals.shtml) to decrease the number of clusterheads produced in the original LCA and to decrease the number of clusterheads generated in the pathological case. In LCA2, a node is said to be covered if it is in the 1-hop neighbourhood of a node that has declared it to be a clusterhead.

Starting from the lowest id node to the highest id node, a node declares itself to be a clusterhead if among the non-covered nodes in its 1-hop neighbourhood, it has the lowest id. So, LCA2 sometimes favours lower id nodes also during election of clusterhead. WCA (Chatterjee et al., 2002) elects clusterheads based on degree of nodes, the cumulative time during which the node has acted as a clusterhead and its running average of speed till current time. It is based on the heuristics that a node which has already served as a clusterhead for a long time or has a huge number of neighbours is expected to have depleted a significant amount of available battery power. Moreover, if a node is highly mobile, then it may easily loose connection with its neighbours. So, for a node to be clusterhead, it should be less mobile and highly powerful. Nodes are assigned weights as a combination of the afore-mentioned parameters. The weight is directly proportional to mobility and inversely proportional to remaining power. The node with smallest weight is chosen as clusterhead. The main objective of ‘min-max d-cluster algorithm’ (Yu and Chong, 2005) is to divide the network into clusters designed as connected dominating sets where each cluster member is up to d hops away from the clusterhead where d>0. For election of clusterhead, each node initiates 2d rounds of flooding and maintains a logged entry of the results of each flooding round. The rounds are segmented into 1st d rounds and 2nd d rounds. The 1st d rounds propagate the largest node ids. After completion of the 1st d rounds of flooding, 2nd d rounds of flooding begin, using the values that exist at each node after 1st d rounds. The 2nd d rounds propagate smaller node ids to reclaim some of the territory. After completion of the 2nd d rounds, each node looks at its logged entries for the 2d rounds of flooding.

The following rules explain the logical steps of the heuristic that each node runs on the logged entries.

Rule 1

Each node checks to see if it has received its original node id in the second d rounds of flooding. If it has, then it can declare itself a clusterhead and skip the rest part of the heuristic; otherwise, rule 2 is applied.

Rule 2

Each node looks for node pairs. Once a node has identified all node pairs, it selects the minimum id node pair to be the clusterhead. If a node pair does not exist for a node, then rule 3 is applied.

Rule 3

The maximum node id in the 1st d rounds of flooding is elected as clusterhead for the underlying node. The clustering algorithm avoids clock synchronization overhead providing additional processing savings. Furthermore, the number of messages sent from each node is limited to a multiple of d, the maximum number of hops away from the clusterhead. Additionally, because d is an input value, there is control over number of clusterheads elected or density of clusterheads in the network. The amount of resources required at each node is minimal, consisting of three simple rules and two data structures. Nodes are candidates to be clusterheads based on their node id only. In ‘stable clustering algorithm’ (Sheu and Wang, 2006), Shen et al. (?) set up a battery power level as threshold and define nodes whose battery level is below the threshold as bottlenecks. SCA instructs each node to have a count of the number of neighbours that are bottlenecks. The node with largest number of bottlenecks is elected as clusterhead. By taking detour in the election, nodes with the least battery power are kept away from becoming clusterheads. Thus, the clusters are expected to be more stable. Unfortunately, because the mobility of nodes is not considered in the election, the possibility of re-clustering is still high when elected cluster heads have high mobility.

In ‘robust clustering algorithm’ (http://www.cs.ou.edu), to overcome the negative effects caused by nodes moving fast or moving back and forth, the average connection time (ACT) of each node with its neighbours during a time period is introduced as the major parameter to form and maintain clusters. Nodes having the largest ACT value become clusterheads. However, the concept of ACT is similar to the cumulative time in WCA or elapsed time, it cannot accurately reflect the current level of battery power because a node may have been connected with its neighbours for too long and it may almost run out of battery power. Topology adaptive clustering algorithm (Chinara and Rath, 2009) is a distributed algorithm that takes into account the mobility of a node and its available battery power as the parameters to determine the clusterhead. The average of last few displacements gives the average speed of any node. Thus the difference of maximum permissible speed of a node and average speed gives the mobility factor of a node. A large mobility factor indicates a slower node and a small mobility factor indicates a faster node. Available battery power is the energy contained in the node at the instant of weight calculation. These two parameters are added with different weight factors to find weights of individual nodes. The node with highest weight is elected as clusterhead. Bird flight inspired clustering (BFIC) (Tiwari et al., 2010) is inspired by bird flight where birds travel long distances in flocks which are ‘V’ shaped, where the entire burden is on the sphere head of the flock due to the increase in the induced drag on the sphere head because of the down wash of the appositely rotating line vortices. But, this down wash on the sphere head also has a positive effect of up wash which reduces the up thrust required by the rest of the birds in the flock. Thus, it reduces the amount of energy required by the birds behind the wing of the sphere head to fly. This is the reason why birds travel long distances by loosing minimum amount of energy. This concept is used in BFIC.

It follows a three tier architecture model consisting of clusterheads in the 1st layer, gateway nodes in the 2nd and ordinary members in the last one. Election of clusterheads depends upon the energy and number of neighbours of the node. A multi-hop clustering algorithm based on neighbourhood benchmarks (MCNB) is proposed in http://dpse.eas.asu.edu. This article assumes that all network links are bidirectional. The score s_i of a mobile node n_i, used to indicate qualification of the node to be a clusterhead is defined as:

$s_i = d_i/lf_i$

Where d_i is the neighbourhood degree of n_i and lf_i is the number of link failures encountered by n_i in unit time, indicating link stability of its neighbourhood.

A node is attached to a cluster provided distance of the new node from head of the cluster is less than or equal to R. R is a pre-determined constant less than or equal to the hop count in the network. In this paper, we propose a fuzzy-controlled multi-hop adaptive clustering (FMAC) for ad hoc networks where two fuzzy controllers are embedded in each node – CHE (cluster head elector) and CAE (cluster attachment evaluator). These fuzzy controllers consider various parameters of nodes like remaining energy, rate of energy depletion, relative velocity with the neighbours, cardinality of the set of neighbours, etc. The rule bases are based on some heuristics written as follows:

1) If a node has high remaining battery life and low rate of energy depletion, then its claim to become a clusterhead gains strength from the point of view of energy dependent sustainability.
2) If a node has low relative velocity with respect to its neighbours, the possibility of survival of the wireless links from the node to its neighbours increase. This is extremely desirable for a clusterhead.
3) If the wireless link between a cluster head and cluster member survives for a long time without break then it has high chance of survival in near future.
4) The more cluster members stay within the radio-range of the clusterhead, the better. This will reduce the delay to broadcast information within the cluster by the clusterhead. Hence, it is good for a node that wants to become a clusterhead, to have a high radio-range.
5) As far as the possible attachment between a node n_i and a clusterhead is concerned, the node n_i should be connected to the cluster through a node n_j s.t. both n_i and n_j have sufficient remaining lifetime and low velocity relative to one another.

The observations expressed earlier are in the form of if-then rules which are the basic unit of fuzzy function approximation. Advantages of fuzzy logic are that it is flexible, conceptually easy to understand and based on natural language. Moreover, it is tolerant of imprecise data and can model non-linear functions of arbitrary complexity. All these encouraged us to design the scheme of FMAC using fuzzy logic.

ELECTION OF CLUSTERHEAD

Here, we allocate weight to every node in a cluster. The weight depends upon its remaining lifetime, transmission range, affinity with neighbours in terms of relative velocity and proximity. In FMAC, each node transmits HELLO message at regular intervals. The attributes of HELLO message transmitted by n_i at time t contains the following information about it:

i) Node identification number n_i,
ii) Radio-range R_i,
iii) Geographical position $(x_i(t), y_i(t))$ of n_i at time t in terms of latitude and longitude,

iv) Velocity $v_i(t)$ of n_i at time t,
v) Timestamp t,
vi) Clusterhead status (set to 1 if n_i is a clusterhead, otherwise it is set to 0).

After receiving the HELLO message of n_i, its neighbours reply with the ACK (acknowledgement) message. The attributes of ACK message transmitted by a neighbour n_j of n_i at time t consists of the following information:

i) Source identification number n_j,
ii) Destination identification number n_i,
iii) Velocity $v_j(t)$ of n_j at time t,
iv) Geographical position $(x_j(t), y_j(t))$ of n_j at time t in terms of latitude and longitude,
v) Identification number of heads of those clusters of which n_j is a member.

In case of change of clusterhead, a head_change message is flooded within the respective cluster. The attributes of head_change are as follows:

i) Identification number of the new clusterhead,
ii) Timestamp.

In case a node n_k, which is a member of a cluster CLK, detects that a node n_p has left its neighbourhood, then it sends a member_lost message to the clusterhead n_i of CLK with the following attributes:

i) Identification number n_p of the lost member,
ii) Identification number of the node n_k through which n_p was connected to CLK,
iii) Destination identification number n_i,
iv) Timestamp.

In case a new node n_p joins a cluster CLK through a cluster member n_k, then n_k sends a member_join message to the clusterhead n_i of CLK with the following attributes:

i) Identification number n_p of the new member,
ii) Identification number of the node n_k through which n_p has been connected to CLK,
iii) Destination identification number n_i,
iv) Timestamp.

After joining the network, a node can start to elect clusterhead after a pre-determined time interval T_INV. If an isolated node (not connected to any cluster) receives HELLO messages from any clusterhead before T_INV, it accepts membership of that cluster unconditionally. On the other hand, if a cluster member receives HELLO message from any clusterhead, then it joins the cluster provided its CAE permits and the distance of the new member from the clusterhead is less than or equal to a predefined number of hops. Otherwise, it forms a 1-hop cluster with its neighbours provided its set of neighbours is non-empty and its CHE permits.

Parameters of cluster-head elector (CHE)

The input parameters of the clusterhead elector (CHE) are as follows:

Residual energy quotient

The residual energy quotient $\alpha_i(t)$ of a node n_i at time t is defined as:

$$\alpha i(t) = \alpha 1i(t) \exp \alpha 2i(t) \quad (1)$$

Where $\alpha 1_i(t)$ and $\alpha 2_i(t)$ are termed as present energy quotient and energy depletion quotient respectively, and exp stands for exponentiation.

$$\alpha 1i(t) = 1 - ei(t)/Ei \quad (2)$$

$e_i(t)$ and E_i indicate the consumed battery power at time t and maximum or initial battery capacity of n_i, respectively. It may be noted from the formulation in Equation 2 that:

$$0 \leq \alpha 1_i(t) \leq 1$$

Values close to 1 enhance capability of n_i as a router. Let, n_i start operating in the network at time t_i with battery capacity E_i. Then, the rate of energy depletion of n_i at time t is $e_i(t)/(t - t_i)$. Energy depletion quotient $\alpha 2_i(t)$ is mathematically expressed as:

$$\text{So, } \alpha 2_i(t) = \begin{cases} e_i(t)/(t - t_i) & \text{if } e_i(t) \leq (t - t_i) \\ 1 - (t - t_i)/e_i(t) & \text{otherwise} \end{cases} \quad (3)$$

$\alpha 2_i(t)$ also lies between 0 and 1. The lesser the energy depletion quotient, the better for the associated node.

Head density quotient

The head density quotient $\beta\tilde{}_i(t)$ of clusterhead n_i at time t is given by:

$$\beta\tilde{}_i(t) = \{\Pi \quad \beta_{ij}(t)\} \exp (1/|NE_i(t)|) \quad (4)$$

$$n_j \in NE_i(t)$$

Where $\beta_{ij}(t)$ is formulated in Equation 5 and $NE_i(t)$ is the set of neighbours of n_i at time t.

Affinity $\beta_{ij}(t)$ of the link between the nodes n_j and its predecessor clusterhead n_i is defined in Equation 5 where n_j has been continuously residing within neighbourhood of n_i from $(t-\varpi_{ij}(t))$ to current time t.

$$\beta_{ij}(t) = \begin{cases} 0.1 & \text{if } \varpi_{ij}(t) \le \Gamma_{min} \\ 1 & \text{if } \varpi_{ij}(t) > \Gamma_{max} \\ \left(\dfrac{\varpi_{ij}(t) - \Gamma_{min}}{\Gamma_{max} - \Gamma_{min}}\right) f1_{ij}(t)\, f2_{ij}(t) & \text{otherwise} \end{cases} \quad (5)$$

Where $f1_{ij}(t) = \{1 - (|\nu_i(t) - \nu_j(t)| + 1)^{-1}\}$ and $f2_{ij}(t) = \{1 - d_{ij}(t) / (R_i + 1)\}$

Since the minimum length of a multi-hop path in an ad hoc network is 2, minimum delay Γ_{min} for multi-hop communication is given by:

$$\Gamma_{min} = 2\, R_{min}/v$$

Where v is speed of the wireless signal and R_{min} is the minimum available radio-range in the network.

Assuming H to be the maximum allowable hop count in the network, maximum number of routers in a communication path is (H - 1). If τ denotes the upper limit of waiting time of that packet in message queue of any node and R_{max} denotes the maximum available radio-range in the network, maximum delay Γ_{max} for multi-hop communication is given by:

$$\Gamma_{max} = H\, R_{max}/v + (H - 1)\, \tau$$

In the worst case delay or maximum delay situation, a packet has to traverse the maximum available number of hops that is H with length of each hop being the maximum possible that is R_{max}. Hence the total distance traversed by the wireless signal in its worst case journey from source to destination is HR_{max}. The signal velocity is v that is a packet can traverse v unit distance in unit time. Hence the time required to travel the distance of HR_{max} is HR_{max}/v. This is the upper limit of travelling time for a packet. Also the waiting time in routers are involved in worst case. Maximum age of a packet in message queue of a router is assumed to be τ and H – 1 is the highest possible number of routers in a path. So, the upper limit of waiting time of a message throughout its journey from source to destination is $(H - 1)\tau$. The maximum delay Γ_{max} for multi-hop communication is actually the sum total of the upper limits of the afore-mentioned travelling time and waiting time for a packet. In the aforementioned formulation, $\nu_i(t)$ specifies velocity of node n_i at time t. $d_{ij}(t)$ and R_i signify the distance between n_i and n_j at time t and radio range of n_i, respectively. All other symbols carry their usual meaning. The situation $\varpi_{ij}(t) \le \Gamma_{min}$ indicates that either n_j is completely new as a neighbor to n_i or n_j did not steadily reside within the neighborhood of n_i even for a time interval so small as Γ_{min}. Hence the link stability is negligible denoted by 0. On the other hand, if $\varpi_{ij}(t) > \Gamma_{max}$, it indicates that n_j has been continuously residing within the neighborhood of n_i for more than the time span that may be required at most for a message to traverse from its source to destination.

In this situation the stability is 1. Otherwise, the ratio $(\varpi_{ij}(t) - \Gamma_{min})/(\Gamma_{max} - \Gamma_{min})$ is used to predict future of the neighborhood relation between n_i and n_j based on its history so far. If $\varpi_{ij}(t)$ is close to Γ_{min}, the ratio $(\varpi_{ij}(t) - \Gamma_{min})/(\Gamma_{max} - \Gamma_{min})$ takes a small fractional value. Similarly, it is evident from Equation 5 that as $\varpi_{ij}(t)$ approaches Γ_{max}, the value of the aforementioned ratio proceeds towards 1. Relative velocity of n_i with respect to (w.r.t) n_j at time t is given by $(\nu_i(t) - \nu_j(t))$. Its effect on $\beta_{ij}(t)$ is modeled as $f1_{ij}(t)$. Please note that $f1_{ij}(t)$ always takes a fractional value between 0 and 1, even when $\nu_i(t) = \nu_j(t)$. As the magnitude of relative velocity of n_i w.r.t. n_j at time t increases, it leads to the decrease in value of $f1_{ij}(t)$, which in turn, contributes to increase the link stability. $f2_{ij}(t)$ expresses the dependence of $\beta_{ij}(t)$ on the distance between the nodes n_i and n_j at time t. Since n_i is the predecessor of n_j at time t, n_j must be within the transmission range (or radio-range) of n_i at that time. Since R_i denotes the radio-range of n_i, upper limit of the distance $d_{ij}(t)$ between n_i and n_j at time t is R_i. As per the expression of $f2_{ij}(t)$, it also acquires a fractional value between 0 and 1. As $d_{ij}(t)$ increases, $f2_{ij}(t)$ decreases enhancing the link stability. Note that, $\beta_{ij}(t)$ always ranges between 0 and 1.

Radio quotient

Radio quotient $RQ_i(t)$ of node n_i at time t is expressed in Equation 6:

$$RQ_i(t) = (R_i - R_{min})/(R_{max} - R_{min} + 1) \quad (6)$$

It is evident from Equation 6 that radio quotient lies between 0 and 1. 1 is introduced in the denominator in order to avoid 0/0 situation where $R_{max} = R_{min}$. The higher the radio-quotient of a clusterhead, the better for the cluster.

Rule bases of CHE

The parameters of CHE are divided into crisp ranges and the corresponding fuzzy variables are shown in Table 1. According to the study of discharge curve of batteries heavily used in ad hoc networks, at least 40% (fuzzy variable a1 represents the range 0 to 0.40) of total charge is required to remain in operable condition; 40 to 60% (fuzzy variable a2) of the same is satisfactory, 60 to 80% (fuzzy variable a3) is good and the next higher range (that is 80 to 100% or fuzzy variable a4) is more than

Table 1. Crisp ranges of parameters and fuzzy variables.

Crisp ranges of α and $\tilde{\beta}$	Crisp ranges of RQ	Fuzzy variable
0 - 0.40	0 - 0.25	a1
0.40 - 0.60	0.25 - 0.50	a2
0.60 - 0.80	0.50 - 0.75	a3
0.80 - 1.00	0.75 - 1.00	a4

Table 2. Fuzzy combination of α and β producing output t1.

β \ α	a1	a2	a3	a4
a1	a1	a1	a1	a1
a2	a1	a2	a2	a2
a3	a1	a2	a3	a3
a4	a1	a2	a3	a4

Table 3. Fuzzy combination of t1 and RQ producing output wt.

RQ \ t1	a1	a2	a3	a4
a1	a1	a2	a3	a3
a2	a1	a2	a3	a3
a3	a1	a2	a3	a4
a4	a1	a2	a3	a4

sufficient from the perspective of the remaining energy. Range division of α follows the steps of present energy quotient in the worst case scenario where the energy depletion quotient $\alpha 2$ is equal to 1. High head density is as indispensable as battery power in the process of formation of a strong bonding between a clusterhead and its neighbours. Hence $\tilde{\beta}$ follows the same range distribution of α. RQ follows uniform range distribution between 0 and 1 that is (0 to 0.25 as a1, 0.25 to 0.50 as a2, 0.50 to 0.75 as a3 and 0.75 to 1.00 as a4). Table 2 combines the effects of α and $\tilde{\beta}$ in determination of weight of a clusterhead. Both are given equal importance.

The output produced by Table 2 is t1. The fuzzy composition of t1 and RQ appears in Table 3. In this table, t1 is assigned more importance than RQ because t1 is a composition of two parameters both of which are more important than RQ. For a node to become clusterhead, its weight should be either a3 or a4.

CLUSTER ATTACHMENT DETERMINATION

Because of node mobility in ad hoc networks, the network topology changes with time. A node may join or leave an existing cluster at any time. If a clusterhead does not receive ACK or acknowledgement message from a member after sending HELLO message twice, then the clusterhead concludes that the member has left the cluster. If a clusterhead n_j comes within the radio range of another clusterhead n_i, then the cluster of n_j is included within the cluster of n_i provided the distance of the farthest member of cluster of n_j from n_i is less than or equal to hlim. Hlim is a predefined constant less than H where H is the hop count of the network. As far as inclusion of nodes within a cluster is concerned, we have imposed a constraint here that at most clim number of nodes are permitted to stay within a cluster. A complex situation arises when two clusterheads n_i and n_j come within the radio-ranges one another. In that case, n_j and the cluster members of n_j are included as members of the cluster of n_i provided the following conditions satisfy:

i) The number of members belonging to the cluster of n_j is lesser than the number of members of the cluster of n_i and result of their summation should be less than or equal to clim.

ii) Distance of the farthest member of the cluster of n_j

Table 4. Fuzzy combination of $\alpha_i(t)$ and $\alpha_j(t)$ producing output t2.

$\alpha_i(t)$ \ $\alpha_j(t)$	a1	a2	a3	a4
a1	a1	a1	a1	a1
a2	a1	a2	a2	a2
a3	a1	a2	a3	a3
a4	a1	a2	a3	a4

Table 5. Fuzzy combination of t2 and β producing output t3.

t2 \ $\beta_{ij}(t)$	a1	a2	a3	a4
a1	a1	a1	a1	a1
a2	a1	a2	a2	a2
a3	a1	a2	a3	a3
a4	a1	a2	a3	a4

from n_i, in terms of number of hops should be less than hlim.

We have decided a threshold THD on the upper limit of the number of clusters to which a node may belong as ordinary member. If a node n_j isolated or cluster member of single or multiple clusters) comes within the radio-range of another node n_i where n_i is member of a cluster with head n_k, then n_j is included as member of the circle of n_k provided the following conditions satisfy:

i) The number of clusters to which n_j belongs before its possible inclusion in the cluster of clusterhead n_k is less than THD.
ii) The distance of n_j from n_k in terms of number of hops is less than hlim.
iii) Total number of nodes in the cluster should be less than clim.
iv) The connectivity of n_j to the cluster of n_k as determined by the fuzzy controller CAE (cluster attachment evaluator) is either a3 or a4.

Parameters of CAE

Let it be that an isolated node n_j has arrived within the radio-range of a node n_i which is a member of a cluster CLT, n_j will join CLT provided its attachment with the cluster CLT as evaluated by the CAE of n_j is either a3 or a4. The parameters of CAE are $\alpha_i(t)$, $\alpha_j(t)$, $\beta_{ij}(t)$, $RQ_i(t)$ and hop count quotient from head of CLT to n_j. The definition of hop count quotient is as follows:

Hop count quotient

Assuming that the number of hops from the head of CLK to n_j be h_j, the hop count quotient HP_CNT(j) from head of CLK upto node n_j is given by:

$$HP\_CNT(j) = h_j/H \quad (7)$$

Hop ratio of any route lie between 1/H and 1. Values close to 1/H increase agility of communication from clusterhead to cluster members. Hop count quotient ranges as: ((1/H) – ¼(1 + 3/H)) as a1, (¼ (1 + 3/H) – ½ (1 + 1/H)) as a2, (½ (1 + 1/H) – ¼ (3 + 1/H)) as a3 and (¼ (3 + 1/H) – 1) as a4.

Rule bases of CAE

Table 4 combines $\alpha_i(t)$ and $\alpha_j(t)$ to produce a temporary output variable t2. Both $\alpha_i(t)$ and $\alpha_j(t)$ are assigned equal weightage because survival of both n_i and n_j are indispensable for survival of the link from n_i to n_j. t2 is combined with $\beta_{ij}(t)$ in Table 5 producing t3. Here also both input parameters are given equal importance. t3 and $RQ_i(t)$ are combined in Table 6 generating the next temporary output t4. t3 dominates than $RQ_i(t)$ because t3 is the fuzzy composition of three parameters all of which are more important than $RQ_i(t)$. The fuzzy composition of t4 and HP_CNT(j) is illustrated in Table 7. The output of Table 7 is ct which denotes the cluster attachment or the output of CAE. n_j will join the cluster through n_i provided ct is either a3 or a4.

Table 6. Fuzzy combination of t3 and RQ producing output t4.

t3 / $RQ_i(t)$	a1	a2	a3	a4
a1	a1	a2	a3	a3
a2	a1	a2	a3	a3
a3	a1	a2	a3	a4
a4	a1	a2	a3	a4

Table 7. Fuzzy combination of t4 and HP_CNT(j) producing output t4.

t3 / HP_CNT(j)	a1	a2	a3	a4
a1	a2	a2	a3	a4
a2	a1	a2	a3	a4
a3	a1	a2	a3	a3
a4	a1	a1	a2	a3

ANALYSIS OF FMAC

The overhead of HELLO protocol, cluster formation and cluster maintenance overhead are discussed as follows:

Hello protocol overhead

In order to discover neighbourhood of a node, each node periodically broadcasts HELLO messages. Thus, HELLO protocol introduces an overhead of (f_{hello} * N) packets per time step for all nodes where f_{hello} is the number of HELLO messages broadcast by a node per time step. $f_{hello} = \Theta(1)$ because f_{hello} is proportional to average node speed and inversely proportional to the transmission radius of the associated node and both of these are less than or equal to some predefined constant. The average node speed is limited by the maximum possible node speed v_{max} in the network and transmission range limited by maximum possible radio-range is R_{max}.

Cluster formation overhead

Immediately after each node calculates its weight, it broadcasts its weight to all of its 1-hop neighbours. After receiving weights from all its 1-hop neighbours, each node either becomes a clusterhead or joins a cluster in one time step. Thus, the cluster formation overhead is N messages per time step.

Cluster maintenance overhead

Link breakage between a member and its clusterhead that yields a head change

Consider Figure 1 where n_i is the clusterhead and its neighbours are n_j, n_k and n_l. These three nodes have their own neighbours and so on. The clusterhead n_i maintains information about the best possible path (according to any standard unicast routing protocol) to all cluster members giving it the shape of a tree structure. Also each node maintains the best path to all its successor nodes. For example, in Figure 1, n_d is shown as a neighbour of n_k. It does not mean that n_d is not a neighbour n_j or n_k or some other node. But it denotes that the best path from n_i to n_d is $n_i \rightarrow n_k \rightarrow n_d$. If the link between n_i and n_j breaks, n_j changes its status from ordinary cluster-member to clusterhead provided its set of neighbours is non-empty. On the other hand, if the set of neighbours is empty, then it changes its status from cluster member to isolated node. So, actually two clusters are formed, one with clusterhead n_j and the other with clusterhead n_i with children n_k and n_l. The successors of n_j belong to the cluster of n_j while n_k, n_l and their successors remain in the cluster of n_i. n_j detects breakage of its link with n_i, if it does not receive any HELLO messages from n_i within time interval 3*THELLO where THELLO is the interval between consecutive HELLO messages. After detection of link breakage, n_j will broadcast a head-change message to all of its successors.

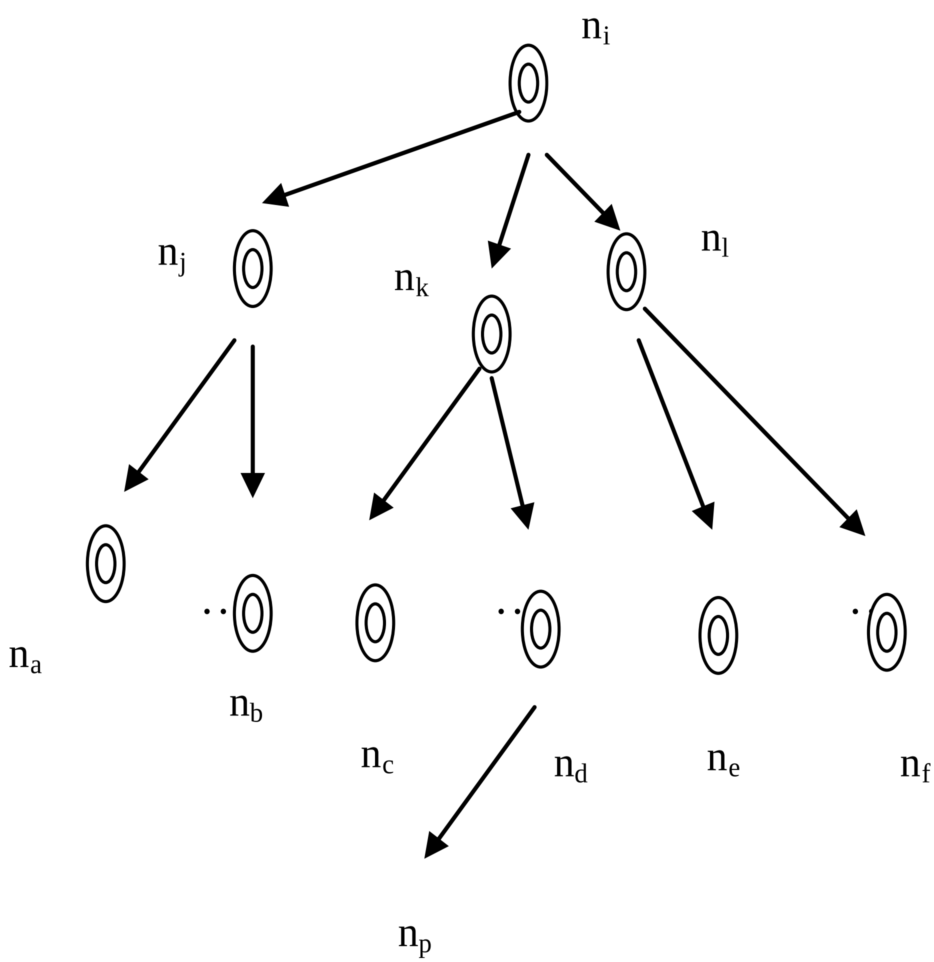

Figure 1. Tree structure of unicast communication of a cluster.

If depth of the tree generating from n_j is dp(j) and the average number of neighbours of a node is denoted as a_n, then the cost C_headchange of head change message is given by:

$$C\_headchange = 1 + a\_n + a\_n^2 + \ldots + a\_n^{dp(j)-1} \quad (8)$$

That is, $C\_headchange = (a\_n^{dp(j)}-1)/ (a\_n - 1)$ (9)

dp(j) is less than or equal to (hlim-1) since, maximum possible distance of any cluster member from n_i, is hlim. So, C_headchange is $O((a\_n^{hlim-1}-1)/ (a\_n-1))$. Value of the hop count H is decided in such a fashion that if a node starts flooding, the flooded message should ideally reach all nodes in the network. Since clim indicates total number of nodes in the cluster, then:

$$clim = (a\_n^{hlim}-1)/ (a\_n - 1)) \quad (10)$$

Hence, from Equations 9 and 10 it can be concluded that C_headchange is O(clim).

Link breakage between a member and its clusterhead that yields a head change

One example of this kind is breakage of the link from n_d to n_p. In this case, n_d informs the clusterhead n_i that n_p is not a cluster member now. The member_lost message propagates from n_d to n_i, that is a maximum of (H - 1) hops. So, the overhead is O(H).

Link establishment s.t. an isolated node n_p enters into the neighbourhood of a cluster member n_d or clusterhead n_i

If the isolated node n_p enters into the neighbourhood of an ordinary cluster member n_d and the evaluated attachment is a3 or a4, then a member-join message propagates from n_d to n_i. The corresponding overhead is O(H). On the other hand, if the isolated node n_p enters into the neighbourhood of n_i, then no message propagates anywhere. Only HELLO and ACK messages are sufficient. Hence, the overhead in this case is O(1).

Link establishment s.t. two clusterheads n_i and n_j become 1-hop neighbours, yielding a head change

Depending upon the cardinality of the set of cluster members of n_i and n_j, any one of them becomes the clusterhead and the other changes its status from clusterhead to cluster member. Without any loss of generality, let us assume that the node n_i becomes the clusterhead and n_j changes its status from clusterhead to ordinary cluster member. The head-change message has to propagate from n_j to all its cluster members. The situation is similar to (i) and the overhead is O(clim).

Link establishment s.t. two clusterheads n_i and n_j become 1-hop neighbours without yielding a head change

In this situation, clusters of n_i and n_j do not merge at all and even their set of neighbours do not change. So, the overhead is 0.

When a cluster head resigns

Successors of the clusterhead become clusterheads themselves provided their set of neighbours is non-empty. Overhead for changing head is a_n*O(clim) that is O(clim). The successors who do not have any neighbours, change their status to isolated nodes.

Trade-off between intra-cluster and inter-cluster communication

Cost of inter-cluster communication increase if the number of clusters increase or the size of clusters decrease. Decrease in the size of clusters will reduce the cost of intra-cluster communication. It is illustrated mathematically as follows: Let the total number of clusters in the network be denoted as cls_num. Then,

$$cls\_num \times clim = N \quad (11)$$

That is, cls_num = N/clim

Cost of inter-cluster communication is given by O(cls_num). Cost of intra-cluster broadcast and unicast communication are O(clim) and O(hlim) respectively. Hlim is less than or equal to the hop count H of the network. If clim is set to $\sqrt{N}$, then cost of both inter-cluster communication and intra-cluster broadcast becomes $O(\sqrt{N})$. On the other hand, if clim is set to $N^{1/3}$, then cost of inter-cluster communication and intra-cluster broadcast are $O(N^{2/3})$ and $O(N^{1/3})$, respectively. So, clim is the handle that is used to obtain a trade-off between the costs of inter-cluster and intra-cluster communication.

SIMULATIION RESULTS

We choose a 500 × 500 unit square as basic simulation setting. A number of $\wp$ nodes are deployed using a random number generator initialized by independent seeds. In simulation runs, value of $\wp$ has been varied from 20 to 1500 (values being 20, 50, 100, 500 and 1500). The transmission range is a random number between 10 and 70. Clim is set to $N^{1/2}$ where N is the number of nodes with values being 20, 50, 100, 500 and 1500 in various simulation runs. a_n is set to 2, 4, 6 and 8 in different simulation runs. As far as velocities of nodes are concerned, random waypoint mobility model has been used in order to compare results of our proposed scheme FMAC with TACA, BFIC and MCNB. Velocity of network elements or nodes varied between 5 units to 20 units per second. The simulation time was set to 900 s and we have implemented FMAC using the ns-2 (www.isi.org/nsnam/ns) simulator. The metrics that are considered for simulation are packet delivery ratio, message cost per node, rate of change of cluster by members, rate of re-election of clusterhead, clustered node ratio and number of clusterheads. Packet delivery ratio specifies the percentage of data packets that successfully arrive at their respective destinations. AODV is used as the underlying unicast routing protocol for all clustering schemes. Message cost per node is computed as the total number of messages transmitted by all nodes divided by the number of nodes.

Rate of change of cluster by members and rate of re-election of clusterhead indicate the occurrence of cluster attachment and detachment by nodes per unit time and number of clusterheads elected per unit time. The other metric that is total number of clusterheads is quite self explanatory.

DISCUSSION

The clustering algorithm FMAC greatly emphasizes on stability of links. A clusterhead maintains stable connectivity with cluster members. Moreover, a node cannot be attached to a cluster through an unstable or weak link with a cluster member. This reduces the phenomenon of link breakage. Also since FMAC is power aware, it increases node longevity and reduces possibility of breakage due to node exhaustion. In an ad hoc network, if the link in path P from a node n_i to another node n_j breaks, then n_i broadcasts a link repair message to discover a suitable route to n_j or any of its successors in path P. Since FMAC faces much less link breakage

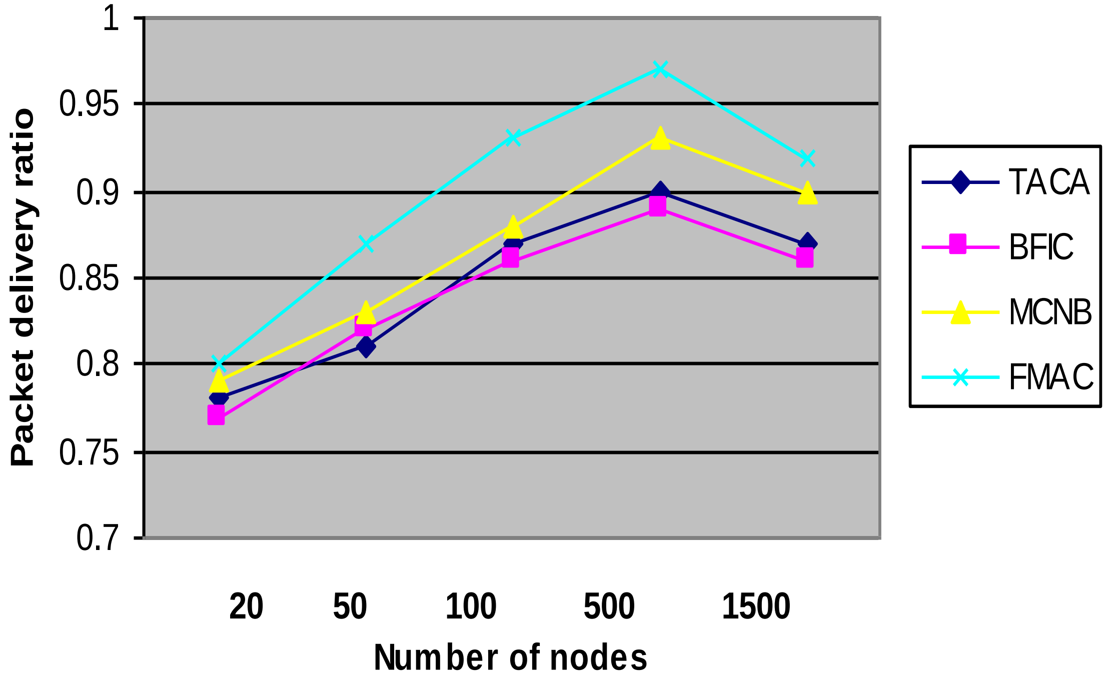

Figure 2. Graphical demonstration of packet delivery ratio versus number of nodes.

compared to other state-of-the-art clustering protocols, message cost due to link breakage is less in FMAC. Please note here that message cost includes also the cost incurred by the clusters during intra-cluster and inter-cluster communication. TACA and BFIC form only single hop clusters whereas FMAC is able to construct big stable clusters. So, all multi-hop communication in TACA and BFIC are inter-cluster which may be intra-cluster in FMAC and MCNB. Intra-cluster communication is less costly because if a cluster member n_i wants to talk with another cluster member n_j at least 2-hop away in the same cluster, then n_i sends the message to the clusterhead. The clusterhead knows stable routes to all the cluster members. So, the clusterhead conveys the message of n_i to n_j. No route discovery process needs to be initiated by n_i to communicate with n_j. On the other hand, in case of single hop clusters, the communication would have been inter-cluster.

In this kind of communication, n_i needs to initiate a route-discovery process. Route discovery means broadcasting route-request messages to the clusters of the network which leads to a high message cost. Since the cost of intra-cluster .communication is lesser than the same of inter-cluster communication and number of intra-cluster communication in FMAC is higher than the same in TACA and BFIC, cost of messages is much lesser in FMAC than its competitors. This is evident from Figure 3. High message cost generates high message contention and collision leading to the decrease in the number of packets that are successfully delivered to the destination. This is illustrated in Figure 2. Also it may be noted from this figure that as the number of nodes increase gradually the packet delivery ratio starts increasing due to the improvement in the number of neighbors of the nodes. But as the number of nodes crosses a threshold or reaches the saturation point, packet delivery ratio starts to reduce because of message contention and collision. Unlike TACA, BFIC and MCNB, clusterheads in FMAC are elected after rigorous analysis of energy, neighbour affinity, radio-range etc., of those nodes. This reduces the rate of re-election of clusterheads (Figure 5). Also, non-isolated (member of some cluster) nodes in FMAC join clusters after rigorously evaluating the goodness of the wireless connection in terms of various parameters while the isolated nodes joins its first cluster without any second thought. So, the rate of change of cluster by nodes is much lesser in FMAC than TACA, BFIC and MCNB whereas the number of clustered nodes with respect to (w.r.t.) the total number of nodes is in FMAC is as good as others. These are quite evident from Figures 4 and 6.

Similarly, in Figure 7, the three mentioned protocols are compared with respect to the average number of clusterheads. The number of clusterheads in FMAC is much less than TACA and BFIC because FMAC constructs multi-hop clusters. MCNB also constructs multi-hop clusters but they are not so stable as in FMAC. Hence, cluster members easily get detached from the clusters and these isolated nodes from new clusterheads till they find a chance (does not matter whether strong or

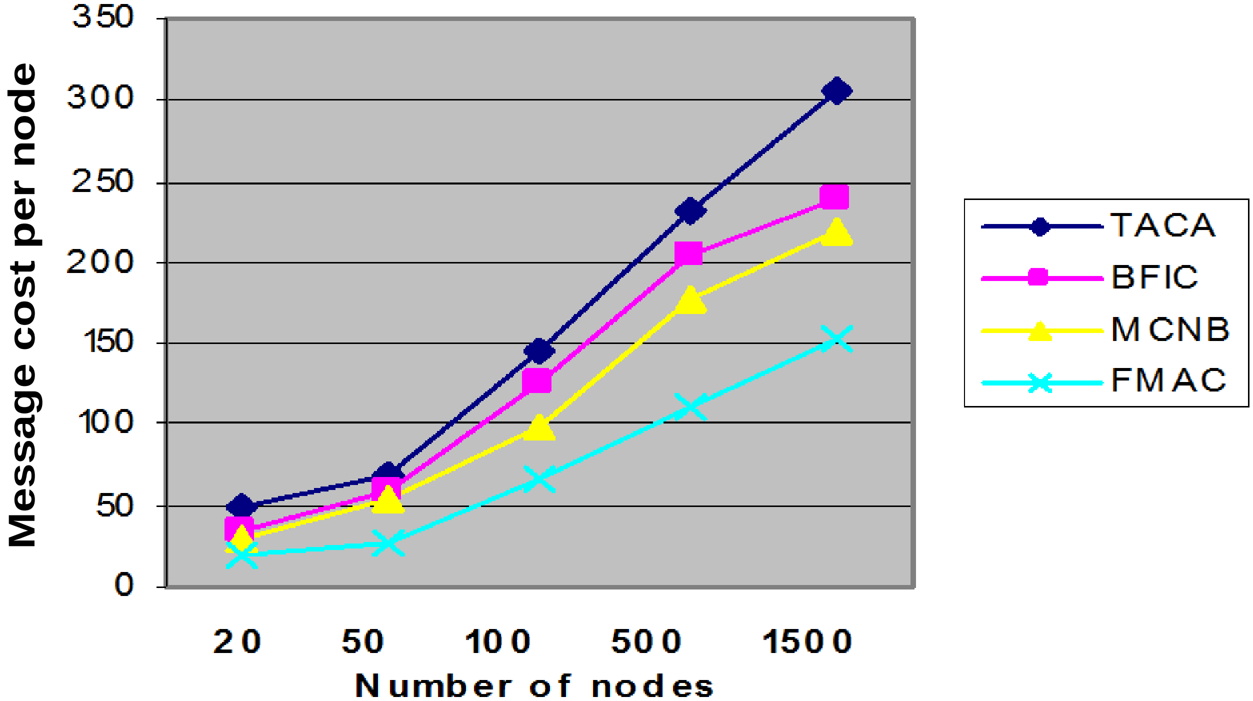

Figure 3. Graphical demonstration of cost of messages per node versus number of nodes.

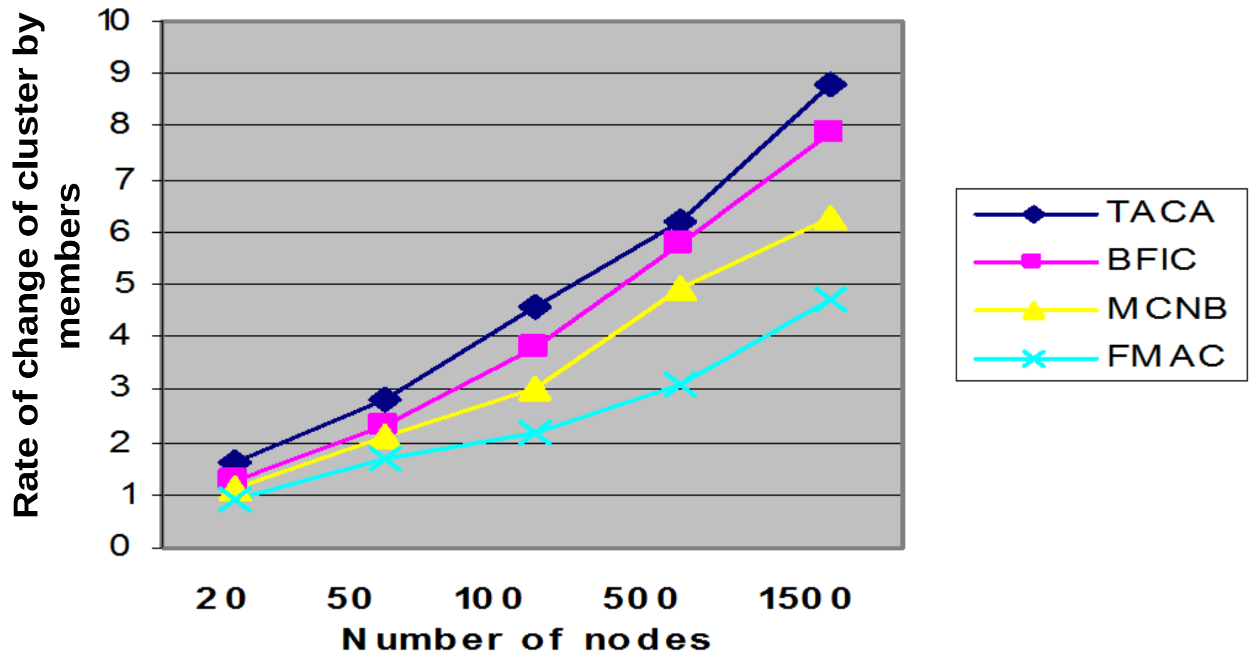

Figure 4. Graphical illustration of rate of change of cluster by members versus number of nodes.

weak) to join a cluster. So, the number of clusterheads in MCNB is higher than FMAC but lesser than the 1-hop clustering protocols TACA and BFIC and also the difference in the number of clusters produced by MCNB and FMAC is smaller than MCNB and TACA as well as BFIC.

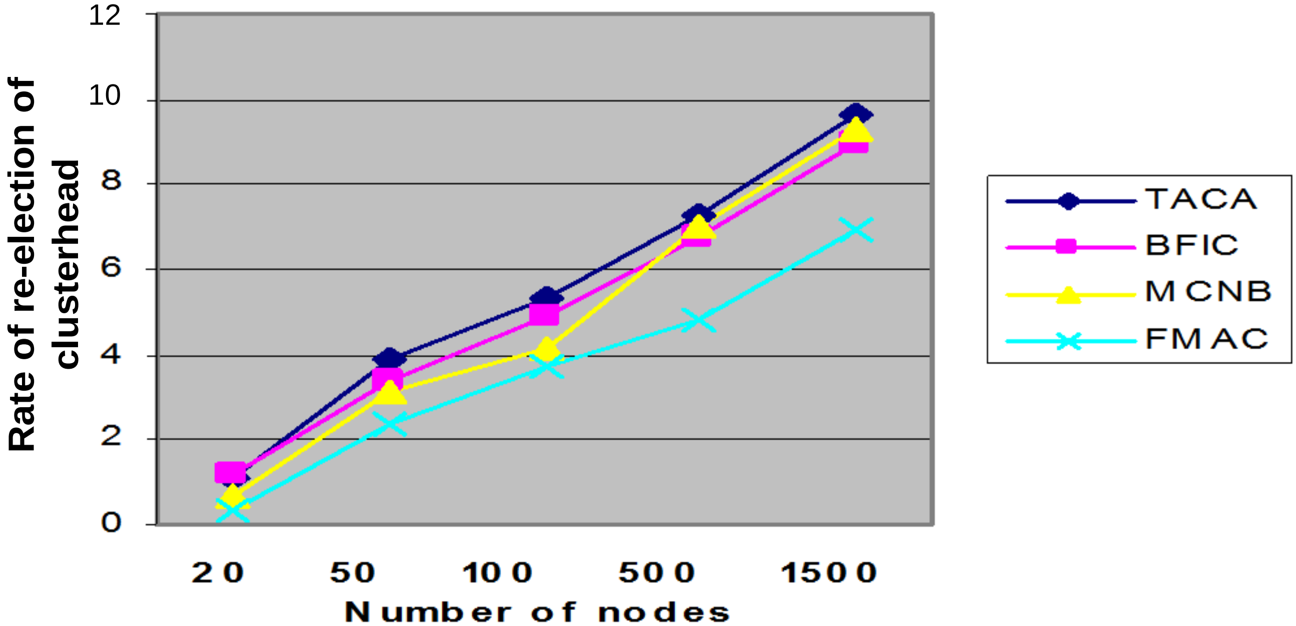

Figure 5. Graphical illustration of rate of re-election of clusterhead versus number of nodes.

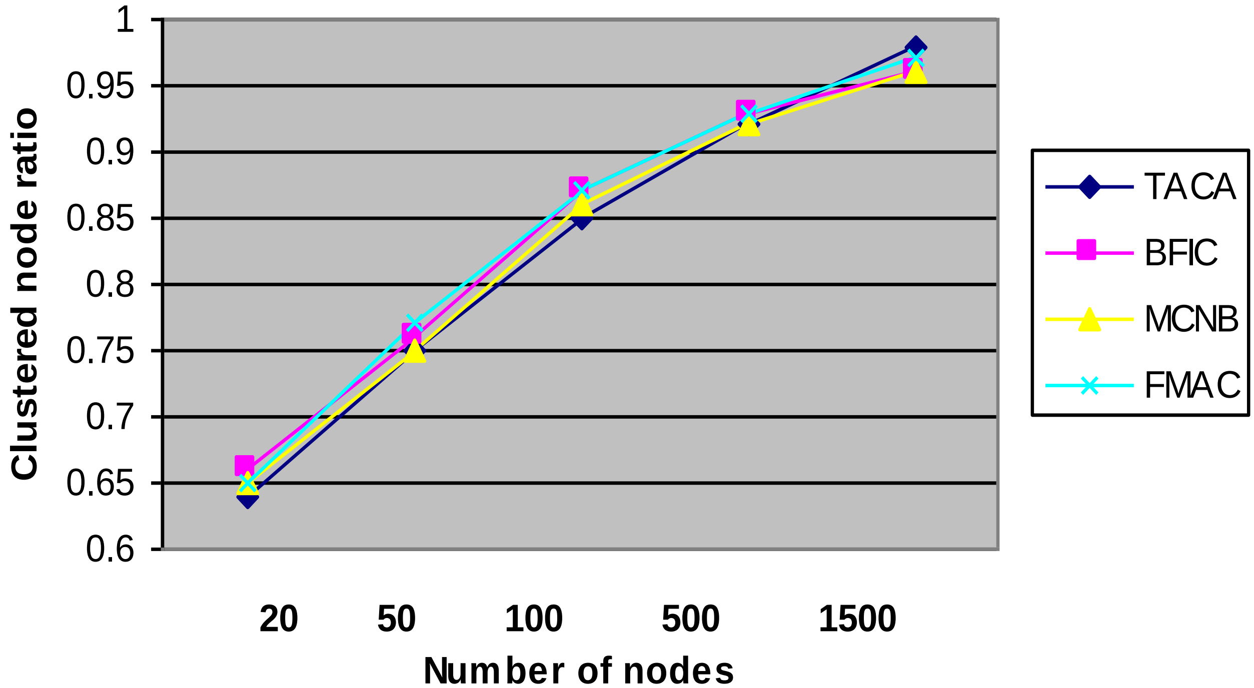

Figure 6. Graphical illustration of rate of clustered node ratio versus number of nodes.

Conclusion

FMAC is a multi-hop clustering scheme that divides the ad hoc network into clusters as much stable as possible. Two fuzzy controllers are embedded in each node to evaluate attachment of the underlying node with a given cluster and to evaluate eligibility of a node as clusterhead. It proposes the trade-off between inter and

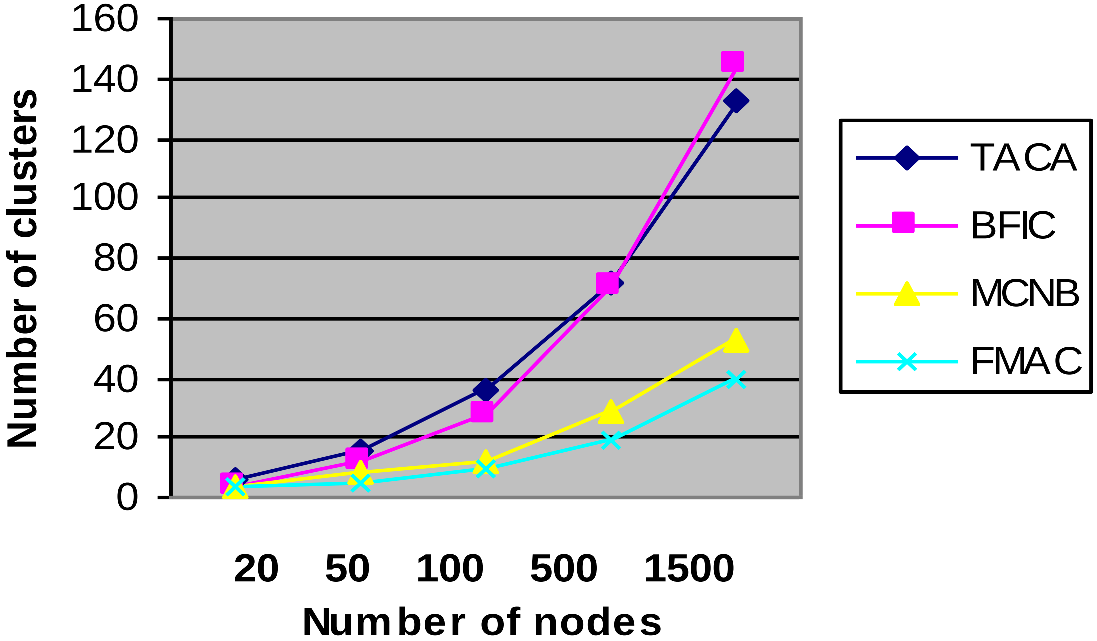

Figure 7. Number of clusters versus number of nodes.

intra-cluster communication by imposing an upper limit constraint on size of clusters. The work can be extended further by applying various unicast protocols other than AODV with FMAC clustering algorithm.

REFERENCES

Agarwal R, Motwani M (2009). Survey of Clustering Algorithm for MANETs, Int. J. Comput. Sci. Eng., pp. 98-104.

Basagni S, Mastrogiovanni M, Panconesi A, Petrioli C (2006). Localized Protocols for Ad Hoc Clustering and Backbone Formation: A Performance Comparison. IEEE Trans. Parallel Distr. Syst., 17(4): 292-306.

Basu P, Khan N, Little T (2001). A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks, in Proceedings of IEEE ICDC 2001, Arizona, USA, pp. 413-418.

Chatterjee M, Das SK, Turgut D (2002). WCA: A Weighted Clustering Algorithm for Mobile Ad Hoc Networks. Cluster Comput., 5(2): 193-204.

Chinara S, Rath SK (2009). TACA: A Topology Adaptive Clustering Algorithm for Mobile Ad Hoc Networks, Proceedings of ICWN 2009, Las Vegas.

Correa B, Ospina L, Hincape RC (2007). Survey of Clustering Techniques for Mobile Ad Hoc Networks, 5(1): 145-161.

http://dpse.eas.asu.edu/tdsm/papers/QShine07.pdf

http://www.cs.ou.edu/~database/documents/Handbook-Chapter-08.pdf

Jane YY, Peter H, Long J (2005). A Survey of Clustering Schemes for Mobile Ad Hoc Networks, 7(1): 102-113.

Sheu P, Wang C (2006). A Stable Clustering Algorithm Based on Battery Power for Mobile Ad Hoc Networks. Tamkang J. Sci. Eng., 9(3): 233-242.

Tiwari M, Varma S (2010). A Bird Fly Inspired Clustering Based Routing Protocol For Mobile Ad Hoc Networks, Int. J. Comput. Sci. Network Secur., 10(3): 141-147.

www.antd.nist.gov/wahn_goals.shtml.

www.isi.org/nsnam/ns.

Yang WD, Zhang GZ (2007). A Weight-based Clustering Algorithm for Mobile Ad Hoc Networks, Proceedings of ICWMC 2007.

Yu JY, Chong PHJ (2005). A survey of Clustering Schemes for Mobile Ad Hoc Networks, IEEE Commun. Surv. Tutor., 7(1): 32-48.