M. REKHA JANAKIRAMAN *et al*, International Journal of Computer Science and Mobile Applications,
Vol.6 Issue. 3, March- 2018, pg. 6-22 **ISSN: 2321-8363**
Impact Factor: 5.515

INTERSECT NODE SELECTION BASED CLUSTERING ALGORITHMS FOR MANET

M. REKHA JANAKIRAMAN[1], Dr. A.SUBRAMANI[2]
[1]M.Phil Scholar, Department of Computer Science, M.V. Muthiah Govt. Arts College for Women, Dindigul, Tamilnadu, India - 624001. E-mail: mail2mrekha@gmail.com
[2]Assistant Professor, Department of Computer Science, M.V. Muthiah Govt. Arts College for Women, Dindigul, Tamilnadu, India - 624001. E-mail: subramani.appavu@gmail.com

***Abstract:** As MANET is functioning in the absence of fixed infrastructure, the packets are sent by self-organization of nodes. So, in the aforementioned definition, self-organization is an important phrase. By self-organizing themselves the nodes route the packets of their neighbor nodes over wireless medium by constructing a multi hop networking environment. So the nodes of the MANET should act as both a router and a transceiver. Since MANET is dynamic in nature, and the nodes have to self-organize to adjust their transmission range to stay connected to each other, the connectivity between the nodes is a big challenge. This network mostly used for emergency network such as military deployment disaster management, etc. The clusterhead and the gate way nodes forward RREQ packets to set up a path between the source and the destination which proves efficient communication, prior to forwarding a RREQ packet cluster the head/gateway node compares its mobility value with RREQ and updates the least value in RREQ. The destination node advertises the least mobility value to the remaining nodes in the path with the help of RREP packet; therefore stable paths are found without increasing the network control overheads. In this paper, we discussed a new algorithm called Intersect Node Selection Based Clustering Algorithm (INSBCA) for MANET in three different scenarios, and its result compare with already existing clustering algorithm*
***Keywords:** Clustering, Node Density, MANET, Routing, Delay.*

1. Introduction

In an ad hoc network, mobile nodes communicate with each other using multihop wireless links. There is no stationary infrastructure; for instance, there are no base stations. Also each node in the network acts as a router, forwarding data packets for other nodes. A research issue in the design of ad hoc networks is the development of dynamic routing protocols that can efficiently find routes between two communicating nodes. The routing protocol must be able to keep up with the high degree of node mobility that often changes the network topology. In a large network, flat routing schemes produce an excessive amount of information that can saturate the network. In addition, given the nodes heterogeneity, nodes may have highly variable amount of resources, and this produces a hierarchy in their roles inside the network. Nodes with large computational and communication power, and powerful batteries are more suitable for supporting the ad hoc network functions (e.g., routing) than other nodes. Cluster-based routing is a solution to address nodes heterogeneity and limit the amount of routing information that propagates inside the network. The idea behind clustering is to group the network nodes into a number of overlapping clusters.

Clustering makes a possible hierarchical routing in which paths are recorded between clusters instead of nodes. This increases the routes lifetime, thus decreasing the amount of routing control overhead. Inside the cluster, one node that coordinates the cluster activities is clusterhead (CH). Inside the cluster, there are ordinary nodes that also have direct access only to this one clusterhead, and gateways. Gateways are nodes that can hear two or more clusterheads. Ordinary nodes send the packets to their clusterhead that either distributes the packets inside the cluster or (if the destination is outside the cluster) forwards them to a gateway node to be delivered to other clusters. By replacing the nodes with clusters, existing routing protocols can be directly applied to the network. Only gateways and clusterheads participate in the propagation of routing control/update messages. In dense networks this significantly reduces the routing overhead, thus solving scalability problems for routing algorithms in large ad hoc networks.

2. Related Works

A study on the clustering scheme for node mobility in mobile ad hoc network has been done and discussed by Cha et al. (2017). A comparative survey of computation of clusterhead in MANET has been made and discussed by Mohd et al. (2015).Weighted Clustering Algorithms in MANET: A survey have been made and analyzed by Amit Gupta et al. (2015). Cluster Head Selection Schemes for WSN and MANET: A survey has been done and discussed by Faraz Ahsan et al. (2013). Performance comparison of routing protocol in MANET has been made and analyzed by Prabu et al. (2012). An Efficient Cluster-Based Routing Algorithm in Ad hoc Networks with Unidirectional Links has been proposed by Su et al. (2008). A hierarchical classification of various clustering schemes for Manet has been made and discussed by Renu Popli et al. (2015). A Survey on Algorithms for Efficient Cluster Formation and Clusterhead selection in MANET has been made and discussed by Mrunal Gavhale et al. (2015). Forecast Weighted Clustering in MANET has been proposed by Piyalikar (2016). An efficient fault tolerance quality of service in wireless networks using weighted clustering algorithm has been investigated by Rathika (2012). An Enhanced Weighted Clustering Algorithm for MANET has been proposed by Kirti Tiwari (2015). Energy efficient routing in MANET through edge node selection using ESPR algorithm has been proposed by Prabu et al. (2014). Improving Ad hoc Network Performance by using an Efficient Cluster Based Routing Algorithm has been made by Ghaidaa Muttasher Abdulsaheb et al. (2015). Efficient Clustering with proposed Load balancing Technique for MANET has been discussed by Supreet Kaur et al. (2015). Clustering Schemes for Mobile Ad hoc Networks: A Review have been made and discussed by Maheswari et al. (2012). Recent technical reviews, issues and challenges in mobile dense nodes have been made and discussed by Rekha and Subramani (2018).

3. Cluster based routing protocol for MANET

Lowest ID cluster algorithm (LIC) is an algorithm in which a node with the minimum *id* is chosen as a clusterhead. Thus, the *ids* of the neighbors of the clusterhead will be higher than that of the clusterhead. A node is called a gateway if it lies within the transmission range of two or more clusterheads. Gateway nodes are generally used for routing between clusters. Each node is assigned to a distinct *id*. Periodically, the node broadcasts the list of nodes that it can hear (including itself). The Lowest-ID scheme concerns only with the lowest node *ids* which are arbitrarily assigned to numbers without considering any other qualifications of a node for election as a clusterhead. Since the node *ids* do not change with time, those with smaller *ids* are more likely to become clusterheads than nodes with larger *ids*. Thus, the limitation of lowest ID algorithm is that certain nodes are prone to power drainage due to serving as clusterheads for longer periods of time.

Max-Mind-cluster formation algorithm generalizes the cluster definition to a collection of nodes that are up to d-hops away from a clusterhead. Due to the large number of nodes involved, it is desirable to let the nodes operate asynchronously. The clock synchronization overhead is avoided, 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 nearest clusterhead, rather than *n* the number of nodes in the network. This guarantees a good controlled message complexity for the algorithm. Additionally, because *d* is an input value to the heuristic, there is control over the number of clusterhead selected for the density of clusterheads in the network. The amount of resources needed at each node is minimal. Nodes are candidates to be clusterheads based on their node *id* rather than their degree of connectivity. As the network topology changes slightly the node's degree of connectivity is much more likely to change than the node's *id* relative to its neighboring nodes

Highest connectivity clustering algorithm (HCC) The degree of a node is computed based on its distance from others. Each node broadcasts its id to the nodes that are within its transmission range. The node with maximum number of neighbors (i.e., maximum degree) is chosen as a clusterhead. The neighbors of a clusterhead become members of that cluster and can no longer participate in the election process. Since no clusterheads are directly linked, only one clusterhead is allowed per cluster. Any two nodes in a cluster are at most two hops away since the clusterhead is directly linked to each of its neighbors in the cluster. Basically, each node either becomes a clusterhead or remains an ordinary node. This system has a low rate of clusterhead change but the throughput is low. Typically, each cluster is assigned to some resources, which is shared among the members of that cluster. As the number of nodes in a cluster is increased, the throughput drops. The re-affiliation count of nodes is high due to node movements and as a result, the highest degree node (the current clusterhead) may not be re-elected to be a clusterhead even if it looses on neighbor. All these drawbacks occur because this approach does not have any restriction on the upper bound on the number of nodes in a cluster.

K-hop connectivity ID cluster in algorithm (K-CONID) combines two cluster in algorithms: the Lowest- ID and the Highest-degree heuristics. 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. 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 one can 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.

Adaptive Cluster Load Balance Method. In HCC clustering scheme, one clusterhead can be exhausted when it serves too many mobile hosts. It is not desirable and the CH becomes a bottle neck. So a new approach is given. In hello message format, there is an "Option" item. If as ender node is a clusterhead, it will set the number of its dominated member nodes as "Option" value. When as ender node is not a clusterhead or it is undecided (CH or non-CH), "Option" item will be reset to 0.

Adaptive Multihop Clustering is a multihop clustering scheme with load-balancing capabilities. Each mobile node periodically broadcasts information about its ID, Clusterhead

ID, and its status (clusterhead/member/gateway) to others within the same cluster. With the help of this broadcast, each mobile node obtains the topology information of its cluster. Each gateway also periodically exchanges information with neighboring gateways in different clusters and reports to its clusterhead. Thus, a clusterhead can know the number of mobile nodes of each neighboring cluster. Adaptive multihop clustering sets upper and lower bounds (U and L) on the number of cluster members within a cluster that a clusterhead can handle. When the number of cluster members in a cluster is less than the lower bound, the cluster needs to merge with one of the neighboring clusters. In order to merge two clusters into one cluster, a clusterhead always has to get the cluster size of all neighboring clusters. It prevents that the number of cluster members in the merged cluster is over the upper bound. On the contrary, if the number of cluster members in a cluster is greater than the upper bound, the cluster is divided into two clusters.

Mobility-based d-hop Clustering Algorithm partitions an ad hoc network into d-hop clusters based on mobility metric. The objective of forming d-hop clusters is to make the cluster diameter more flexible. This algorithm is based on mobility metric 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 estimate its distance from its neighbors. Strong received signal strength implies closeness between two nodes. This algorithm requires the calculation of five terms: the estimated distance between nodes, the relative mobility between nodes, the variation of estimated distance overtime, the local stability, and the estimated mean distance. 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 packet sat low power acting as a distanced node from its physically close neighbor.

Mobility-Based Metric for Clustering proposes a local mobility metric for the cluster formation process such that mobile nodes with low speed relative to their neighbors have the chance to become clusterheads. By calculating the variance of a mobile node’s speed relative to each of its neighbors, the aggregate local speed of a mobile node is estimated. A low variance value indicates that this mobile node is relatively less mobile to its neighbors. Consequently, mobile nodes with low variance values in their neighborhoods are chosen as clusterhead. For cluster maintenance, timer is used to reduce the change in clusterhead rate by avoiding re-clustering for incidental contacts of two passing clusterheads. However, the mobility behavior of mobile nodes is not always considered in cluster maintenance, so a clusterhead is not guaranteed to bear a low mobility characteristic relative to its members during maintenance phase. This scheme is effective for MANETs with group mobility behavior, in which a group of mobile nodes moves with similar speed and direction, as in high way traffic. Thus, a selected clusterhead can normally promise the low mobility with respect to its member nodes. However, if mobile nodes move randomly the performance may reduce.

Mobility-based Frame Work for Adaptive Clustering: Partition a number of mobile nodes into multihop clusters based on (a,t) criteria. The (a,t) criteria indicate that every mobile node in a cluster has a path to every other node that will be available over sometime period ‘t’ with a probability ‘a’ regardless of the hop distance between them. Cluster framework is based on an adaptive architecture designed to dynamically organize mobile nodes into clusters in which the probability of path availability can be bounded, and the impact of routing overhead can be effectively managed. The purpose of this strategy is to support a more scalable routing infrastructure that is able to adapt to high rates of topological change. This is achieved using prediction of the future state of the network links in order to provide a quantitative bound on the availability of paths to cluster destinations.

Least Cluster Change algorithm (LCC) has a significant improvement over LIC and HCC algorithms as for as the cost of cluster maintenance is considered. Most of protocols executes the clustering procedure periodically, and re-cluster the nodes from time to time in order to satisfy some specific characteristic of clusterheads. In HCC, the clustering scheme is performed periodically to check the “local highest node degree” aspect of a clusterhead. When a clusterhead finds a member node with a higher degree, it is forced to handover its clusterhead role. This mechanism involves frequent re-clustering. In LCC the clustering algorithm is divided into two steps: cluster formation and cluster maintenance. The cluster formation simply follows LIC, i.e. initially mobile nodes with the lowest ID in their neighborhoods are chosen as clusterheads. Re-clustering is event driven and invoked in only two cases:

- When two clusterheads move into there a change of each other, one gives up the clusterhead role.
- When a mobile node cannot access any clusterhead, it rebuilds the cluster structure for the network according to LIC.

Adaptive Clustering for mobile wireless network: ensures small communication overhead for building clusters because each mobile node broadcasts only one message for the cluster construction. In this adaptive clustering scheme, every mobile node keeps its own ID and the ID of its direct neighbors in a set Gi. Each mobile node with the lowest ID in their local area declares to be a clusterhead and set its own ID as its cluster ID (CID).The CID information includes a mobile node’s ID and CID. When a mobile node receives CID information from a neighbor j, it deletes j from its set Gi. If the CID information from j is a clusterhead claim, the mobile node checks its own CID aspect. If its CID is unspecified (it is not involved in any cluster yet or larger than the ID (CID) of j, it sets j as its clusterhead. The process continues till all mobile nodes access some cluster. After cluster formation is completed, clusterheads are no longer used in any further cluster maintenance phase. In the maintenance phase, when a mobile node is finding out that the distance between itself and some node j in the same cluster becomes greater than two hops, it invokes a cluster maintenance mechanism. If node i is a direct neighbor of the node with the highest intra-cluster connectivity in its cluster, it remains in the cluster and removes node j; otherwise, it joins a neighboring cluster. As soon as there is no proper cluster to join, it forms a new cluster to cover itself. Since this mechanism likely forms new clusters but without any cluster elimination or merge mechanisms, the cluster size decreases and the number of clusters increases as time advances. Eventually, almost every mobile node forms a single-node cluster, and the cluster structure disappears.

3-hops Between Adjacent Clusterheads (3-hBAC) This algorithm introduces a new node status, "cluster guest", which means that this node is not within the transmission range of any clusterhead, but within the transmission range of some cluster members. The cluster formational ways begins from the neighborhood of the mobile node with the lowest ID [assuming that it is Mobile Node (MO) in a MANET]. The mobile node with the highest node degree in MO's closed neighbor set is chosen to be the first clusterhead. All the direct neighbors of this clusterhead change status to "cluster member". After the completion of the first cluster, the cluster formation procedure can be performed in parallel in the network. A cluster member or a direct neighbor of any cluster member with status "unspecified" (indicating that it is not included in any cluster yet) are denied serving as a clusterhead. A mobile node, which is not denied clusterhead capability, declares as a new clusterhead when it is with the highest node degree in its neighborhood. When a mobile node finds out that it cannot serve as a clusterhead or join a cluster as a cluster member, but some neighbor is a cluster member of some cluster, it joins the corresponding cluster as a cluster guest.

Passive Clustering Most of the clustering algorithms require all the mobile nodes to announce cluster-dependent information repeatedly to build and maintain the cluster structure, and thus clustering is one of the main sources of control overhead. A clustering protocol that does not use dedicated control packets or signals for clustering specific decision is called passive clustering. In this scheme, a mobile node can be in one of the following four states: initial, clusterhead, gateway, and ordinary node. All the mobile nodes are with 'initial' state at the beginning. Only mobile nodes with "initial" state have the potential to be clusterheads. When a potential clusterhead with "initial" state has something to send, such as a flood search, it declares itself as a clusterhead by piggy backing its state in the packet. Neighbors can gain knowledge of the clusterhead claim by monitoring the "cluster state" in the packet, and then record the Clusterhead ID and the packet-receiving time. A mobile node that receives a claim from just one clusterhead becomes an ordinary node, and a mobile node that hears more claims becomes a gateway. Since passive clustering does not send any explicit clustering-related message to maintain the cluster structure, each node is responsible for updating its own cluster status by keeping a timer. When an ordinary node does not receive any packet from its clusterhead for a given period, its status reverts to "initial".

Load Balancing Clustering (LBC) provide a nearby balance of load on the elected clusterheads. Once a node is elected a clusterhead it is desirable for it to stay as a clusterhead up to some maximum specified amount of time or budget. The budget is a user-defined restriction placed on the algorithm and can be modified to meet the unique characteristics of the system, i.e., the battery life of individual nodes. In this algorithm each mobile node has a variable, virtual ID (VID), and the value of VID is set as its ID number at first. Initially, mobile nodes with the highest IDs in their local area win the clusterhead role. LBC limits the maximum time units that a node can serve as a clusterhead continuously; so when a clusterhead exhausts its duration budget, it resets its VID to 0 and becomes a non-clusterhead node. When two clusterheads move into the reach range, the one with higher VID wins the clusterhead role. When a clusterhead resigns, a non-clusterhead with the largest VID value in the neighborhood can resume the clusterhead function. The newly chosen mobile node is the one whose previous total clusterhead serving time is the shortest in its neighborhood, and this should guarantee good energy level for being anew clusterhead. However, the limitation is that the clusterhead serving time alone may not be a good indicator of energy consumption of a mobile node.

POWER-aware Connected Dominant Set is an energy-efficient clustering scheme which decreases the size of a dominating set (DS) without impairing its function. The unnecessary mobile nodes are excluded from the dominating set saving their energy consumed for serving as clusterheads. Mobile nodes inside a DS consume more battery energy than those outside a DS because mobile nodes inside the DS bear extra tasks, including routing information update and data packet relay. Hence, it is necessary to minimize the energy consumption of a DS. In this scheme Energy level (el) instead of ID or node degree is used to determine whether a node should serve as clusterhead. A mobile node can be deleted from the DS when its close neighbor set is covered by one or two dominating neighbors, and at the same time it has less residual energy than the dominating neighbors. This scheme cannot balance the great difference of energy consumption between dominating nodes (clusterheads) and non-dominating nodes (ordinary nodes) because its objective is to minimize the DS rather than to balance the energy consumption among all mobile nodes. Hence, mobile nodes in the DS still likely deplete their energy at a much faster rate.

Clustering for Energy Conservation assumes two node types: master and slave. A slave node must be connected to only one master node, and a direct connection between slave nodes is not allowed. Each master node can establish a cluster based on connections to slave nodes. The area of a cluster is determined by the farthest distance between the master node and a slave node in the cluster. Master nodes are selected in advance, and can only serve a limited number of slave nodes. The purpose of this scheme is to minimize the transmission energy consumption summed by all master–slave pairs and to serve as many slaves as possible in order to operate the network with longer lifetime and better performance.

In single-phase clustering, initially every master node will pages lave nodes with the allowed maximum energy. For each slave that receives one or multiple paging signals, it always sends an acknowledgment message back to the master from which it receives the strongest paging signal. Since a master node can serve only a limited number of slaves, it first allocates channels for slaves that only receive a single-paging signal from itself. If any free channels remain, other slave nodes, which receive more than one paging signal, are allocated channels in the order of the power level of the paging signal received from the master node. Those slave nodes, which do not receive a channel from a master in the channel allocation phase, are dropped in the further COMMUNICATION phase. This mechanism can reduce the call drop rate by giving priority to those slave nodes that only receive single paging signals in channel allocation. Slave nodes, which receive multiple paging signals, always try to communicate with the nearest master. Each connected master–slave pair communicates with the minimum transmission power in order to save energy. To further lower the call drop rate, double-phase clustering re-pages for slaves do not receive a channel in the first round, in its range. The channel allocation procedure also follows the received signal strength. The limitations of this scheme are that paging process prior to each round of communication consumes a large amount of energy. Master node election is not adaptive, and the method of selecting the master node is not specified.

Entropy-based Weighted Clustering Algorithm [7] In WCA high mobility of nodes leads to high frequency of re-affiliation which increase the network overhead. Higher re-affiliation frequency leads to more recalculations of the cluster assignment resulting in increase of communication overhead. Entropy-based clustering overcomes the drawback of WCA and forms a more stable network. It uses an entropy-based model for evaluating the route stability in ad hoc networks and electing clusterhead. Entropy presents uncertainty and is a measure of the disorder in a system. So it is a better indicator of the stability and mobility of the ad hoc network.

Vote-based Clustering Algorithm is based on two factors, neighbors' number and remaining battery time of every mobile host (MH). Each MH has a unique identifier (ID) number, which is a positive integer. The basic information inside the network is Hello message, which is transmitted in the common channel. Making use of node location information and power information, these algorithms introduce the concept of "vote". The Hello message format is given below. MH_ID item is MH' sown ID and CH_ID item is MH's clusterhead ID, Vote item means MH's vote value, i.e., weighted sum of number of valid neighbors and remaining battery time. Option item is used to realize cluster load balance.

MH_ID	CH_ID	Vote	Option

Hello message format:

Vote = w1x (n/N) + w2x (m/M)---------------------------- (2)

w1; w2: Weighted coefficient of location factors and battery time, respectively,
n: Number of neighbors,
N: Network size or the Maximum of members in a cluster,
m: Remaining battery time,
M: The maximum of battery time remaining battery time.

Each MH sends a Hello message randomly during a Hello cycle. If a MH is a new user to the network, it resets "CH_ID" item. This means that the MH does not belong to any cluster and does not know whether it has neighbor hosts. Each MH counts how many Hello messages that can receive during a Hello period, and considers the number of received Hello messages as its own n. Each MH sends another Hello message, in which "vote" item is set to its own vote value and got from Equation. Recording Hello message during second Hello cycles, each MH knows the sender with highest vote and not belonging to any existing cluster is its clusterhead. It sets its next sending Hello message item "CH_ID" to the clusterhead's ID value. When two or more mobile nodes receive the same number of hello packets, the one who owns the lower ID will get priority. Following this approach, every MH knows its clusterhead ID after two Hello message periods.

Weight-Based Adaptive Clustering Algorithm (WBACA) Drawbacks of WCA algorithm are that all the nodes in the network have to know the weight, so fall the other nodes be for starting the clustering process. This process can take a lot of time. Also, two clusterheads can be one-hop neighbors, which results in the clusters not necessarily being spread out in the network. The clustering approach presented in WBACA is based on the availability of position information via Global Positioning System (GPS). The WBACA considers following parameters of a node for clusterhead selection: transmission power, transmission rate, mobility, battery power and degree. Each node is assigned a weight that indicates its suitability for the clusterhead role. The node with the smallest weight is chosen as the clusterhead. The weight of a node N is defined as:

$$WN = w1*M + w2*B + w3*Tx + w4*D + w5/TR \text{------}(3)$$

Where wl, w2, w3, w4 and w5 are the weighing factors for the corresponding system parameters listed below:-

M: Mobility of the node
B: Batterypower
Tx: Transmission power
D: Degreedifference
TR: Transmission Rate

Connectivity, Energy and Mobility driven weighted Clustering Algorithm (CEMCA) The election of the clusterhead is based on the combination of several significant metrics such as the lowest node mobility, the highest node degree, the highest battery energy and the best transmission range. This algorithm is completely distributed and all nodes have the same chance to act as a clusterhead. CEMCA is composed of two main stages. The first stage consists in the election of the clusterhead and the second stage consists in the grouping of members in a cluster. Normalized value of mobility, degree and energy level is calculated and is used to find the quality (normalizedto1) for each node. The node broadcasts its quality to their neighbors in order to compare the better among them. After this, a node that has the best quality is chosen as a clusterhead. In the second stage the construction of the cluster members set is done. Each clusterhead defines its neighbors at two hops maximum. These nodes form the members of the cluster. Next, each clusterhead stores all information about its members, and all nodes record the clusterhead identifier. This exchange of information allows the routing protocol to function in the cluster and between the clusters.

Protocols	Initiation of Protocol	Working of Protocol	Observation/Result	Protocols
NEIGHBOUR DETECTION PROTOCOL (NDP)	Each node probes the other node present in the neighbor	A node lying within the transmission range of the sender node receives the message	Energy constraint is a major problem in this type of protocol	**NEIGHBOURD ETECTION PROTOCOL (NDP)**
TOPOLOGY ADAPTIVE CLUSTERING ALGORITHM (TACA)	A clusterhead is selected with the node having the maximum weight	When the clusterhead utilizes the energy beyond a certain threshold value then a new clusterhead is selected	The increases in the number of clusterheads increase the length of the communication backbone in term so f number of hops. This may increase the end-to-end delay in communication for the packets	**TOPOLOGY ADAPTIVE CLUSTERING ALGORITHM (TACA)**

Proposed Efficient Protocol				
TRANSMISSIO N RANGE ADJUSTMENT PROTOCOL (TRAP)	The objective of the proposed algorithm is to reduce the number of nodes in the virtual backbone. This is made possible by allowing the isolated clusterheads formed during the execution of TACA to adjust their transmission ranges	The TRAP enables isolated nodes to adjust their transmission ranges so that they become the cluster members of other heads reducing the number of nodes in the virtual back bone	TRAP reduces the delay in communication by reducing the number of clusterheads in the network	**TRANSMISSIO NRANGE ADJUSTMENT PROTOCOL (TRAP)**

Table 1: Comparison of NDPTACA and proposed TRAP protocol

There are many clustering schemes for MANETs available in the literature. To evaluate these schemes, we have to decide on the metrics to use for the evaluation. We summarize the comparison in Table 1. The total number of overheads increase when clusters number is high and CHs change frequently are observed in Table 2. The weight based clustering scheme performs better than ID-Neighbor based, topology based, mobility based and energy based clustering. The weight based clustering scheme is the most used technique for CH election that uses combined weight metrics such as the node degree, remaining battery power, transmission power, node mobility, etc. It achieves several goals of clustering: minimizing the number of clusters, maximizing life span of mobile nodes in the network, decreasing the total overhead, minimizing the CHs change, decreasing the number of re-affiliation, improving the stability of the cluster structure and ensuring a good resources management (minimize the band-width consumption).

Table 2: Comparison of Clustering Schemes

Clustering Schemes	Based on	CHs Election	Cluster Radius	Overlapping Clusters	Clusters Number	CH Change	Cluster Stability	Total Over head
LCA	ID-Neighbor	Lowest ID	One-Hop	Possible	High	Very High	Very Low	High
LCC	ID-Neighbor	Lowest ID	One-Hop	Possible	High	High	Low	High
ACA	ID-Neighbor	Lowest ID	One-Hop	No	High	Moderate	Low	High
Max-Min D-cluster	ID-Neighbor	Node ID	K-Hop	No	High	Moderate	Low	Very High
HCC	Topology	Highest degree	One-Hop	No	High	Very High	Very Low	High
3hBAC	Topology	Highest degree	One-Hop	No	Moderate	Relatively High	Low	Very High
α-SSCA	Topology	Node degree	One-Hop	No	Moderate	Relatively Low	High	Low
Associativity-based Cluster	Topology	Associativity and node degree	K-Hop	Yes	Moderate	Relatively Low	High	Relatively High

MOBIC	Mobility	Lowest mobility	One-Hop	Possible	Relatively High	Low	Relatively High	High
Stability-based mobility prediction	Mobility	Node stability	One-Hop	Yes	Relatively Low	Low	Relatively High	Relatively Low
MPBC	Mobility	Lowest mobility	One-Hop	Yes	Relatively Low	Low	High	Low
Mob D Hop	Mobility	Lowest mobility	K-Hop	No	Low	Low	Very High	Low
Cross-CBRP	Mobility	Node ID and mobility	One-Hop	Yes	Relatively High	Relatively Low	Relatively High	Low
MPGC	Energy	Highest energy	One-Hop	Yes	Moderate	Relatively Low	Relatively High	Relatively High
FWCABP	Energy	Lowest weight	One-Hop	Possible	Low	Low	High	Relatively Low
ECEC	Energy	Highest energy	One-Hop	Yes	Moderate	Low	Relatively High	Relatively Low
FWCA	Weight	A combined weight metric	One-Hop	Possible	Low	Low	High	High
SBCA	Weight	A combined weight metric	One-Hop	No	Low	Low	High	Relatively High
EWBCA	Weight	A combined weight metric	One-Hop	No	Low	Low	Very High	Relatively Low
INSBCA	**Weight with Intersect cluster**	**A combined weight metric**	**One-Hop**	**Yes**	**High**	**Low**	**High**	**Very Low**

4. Proposed Concept

In this section, a proposed new routing algorithm named Insect Node Selection Based Clustering Algorithm (INSBCA) for MANET to find the path from source to destination for hop1, hop2 and so on until reach the destination. The clusterhead selection is based on the weight of each node.

Step1: Calculate the weight for each node within the range or same cluster.

Intersect node Selection Based Clustering Algorithm (INSBCA) selects 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 in capable to cover all the nodes. To ensure that clusterheads will not be over-loaded a pre-defined threshold is used which indicates that the number of nodes each clusterhead can ideally support. WCA selects the clusterheads according to the weight value of each node.

$$Wv = w1\Delta v + w2Dv + w3Mv + w4Pv \text{ ------------------(1)}$$

The node with the minimum weight is selected as a clusterhead. The weighting factors are chosen so that

w1 + w2 + w3 + w4 = 1. Mv is the measure of mobility. It is taken by computing the running average speed of every node during a specified time T. Δv is the degree difference. Δv is obtained by first calculating the number of neighbors 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 Δv is calculated as $|dv - \delta|$ for every node v, where δ is a pre-defined threshold. The parameter Dv is defined as the sum of distances from a given node to all its neighbors. This factor is related to energy consumption since more power is needed for larger distance communications. The parameter Pv is the cumulative time of a node being a clusterhead. Pv is a measure of how much battery power has been consumed.

Step2: Based on weight select the node have highest weight Wt(n) as a primary clusterhead when that cluster does not have clusterhead.

Step3: Select the next highest weight node as a secondary clusterhead.

The Proposed Clustering algorithm executes the following scenario:

Scenario 1: Only one node is present in the cluster

Step1: Detect the element/node.
Step2: Consider that node as clusterhead.
(Or)
Step1: Detect the element/node.
Step2: Ignore that node.
(Or)
Step1: Detect the element/node.
Step2: Check the distance of nearest clusterhead in the nearest cluster or intersect cluster.
Step3: If node is nearer or closer to the any nearest clusterhead, then compare the node weight with the nearest clusterhead. If that node has the highest weight then the node is considered to be clusterhead, which has overlapping properties with previous one.

Scenario 2: No one-hop node present within the same range.

Step1: Ignore all nodes which are not one hop.
Step2: If hop nodes are present ensure that it is connected to at least two nodes.
Step3: Find out the neighbor nodes.
Step4: Neglect the nodes which are at maximum distance.
Step5: Choose the nearest two hop nodes.
Step6: Take the radius of cluster present between two hop nodes.
Step7: If radius is greater than the distance between the hop nodes, then the hop nodes are not in the same range. Then find intersect node for further transaction.
Step8: If radius is less than the distance between the hop nodes, then the hop nodes are in the same range, that nodes belong to same clusterhead.

Scenario 3: Clusterhead is shared

Step1: Find out the edge range of the two clusters.
Step2: If there is no space between two cluster edges, then consider the clusters are intersected.
Step3: Detect the element/node.
Step4: Based on weight select the clusterhead.
Step5: If the intersect node have the highest weight compare to the other node within the cluster, then the intersect node is selected to be a clusterhead for more than one cluster. It is maintained in the minimum overhead.

Intersect Node Selection Based Clustering Algorithm (INSBCA) for Data transferred

// All clusters have the same environment properties
// All nodes have same energy at the beginning level
// Node mobility varies from cluster to cluster
// Number of node within the cluster also varies
//Node mobility in all directions

Step1: Choose the nearest node with the highest energy towards the destination, that node not cross the cluster region in meantime of data transfer from source to that node and also not dead end node or selfish node, it should have public sharing nature with highest degree connected more than two nodes, that node intersect the more than one cluster region is well and good.
Step 2: Primary clusterhead taken the responsibility to transfer the data from source to destination is not necessary. Each and every node takes in account to transfer the date from that node to nearest nodes.
Step 3: In data transferred the clusterhead not taking part, at that time it should maintained the transaction information, it is very helpful to partial transaction or unsuccessful transaction. That time the next neighbor node involves in this transaction.
Step 4: The clusterhead energy level is also monitored. The energy level is comedown at 10% immediately the secondary clusterhead takes the charge simultaneously. It is very helpful to avoid the retransmission and data drops.
Step 5: The same process done in all clusters up to the data transfer from source to destination.

5. Results and Discussion

The experimental analysis of the proposed routing protocol compare to existing routing protocol through Network Simulator (NS-2). IEEE 802.11 is used as the MAC layer protocol. The radio model simulates with a nominal bit rate of 2 Mbps. The broadcast mode with no RTS/CTS/ACK mechanisms is used for all message transmissions, including HELLO, DATA, and ACK messages. Nominal transmission range is 250 m. The radio propagation model is the two-ray ground model. The simulation parameters are listed in Table 3. The network area is confined within 1000 × 1000 m^2. The traffic pattern is CBR (constant bit rate) with a network traffic load of 4 packets and the packet length is 512 bytes. The mobility model used is the Random Waypoint Model. The pause time of the node reflects the degree of the node mobility. The small pause time means intense node mobility and large pause time means slow node mobility. The pause time is maintained as 5 s. The simulation time is 500 s. The mobility of the node is performed by varying the speed from 5 to 25 m/s.

Table 3. Simulation Parameters.

Parameters	Values
Simulation	NS-2
MAC Layer Protocol	IEEE 802.11
Mobility Model	Random Waypoint
Terrain Range	1,000 X 1,000 m2
Transmission Range	250 Meters
Examined routing protocol	CFBEC
Channel Bandwidth	2 Mbps
Speed	10-20 m/s
Application Traffic	CBR
Simulation Time	500 s
Propagation mode	Free space
Data Packet size	512 bytes
Packet rate	2 packets/s
Number of Nodes	20–100

The following performance metrics to evaluate through networks simulation (NS2):

Throughput: Throughput is the ratio of the total amount of data that reaches a receiver from a sender to the time it takes for the receiver to get the last packet. It is measured in bits/sec or packets per second. A high throughput network is desirable. It is calculated by the given equation.

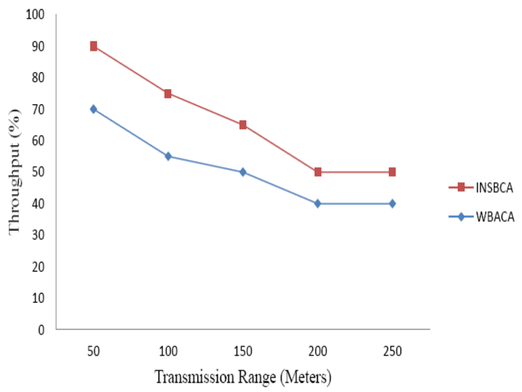

Figure 1: Throughput Vs. Transmission Range.

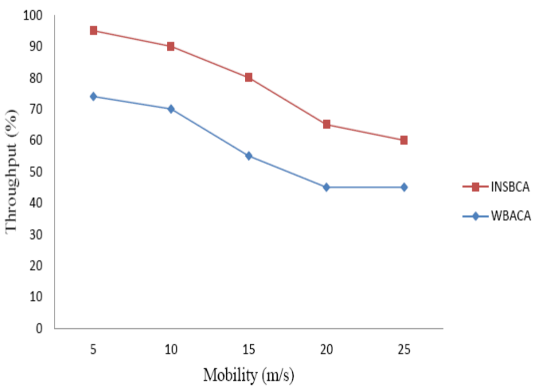

Figure 2: Throughput Vs. Mobility.

M. REKHA JANAKIRAMAN *et al*, International Journal of Computer Science and Mobile Applications,
Vol.6 Issue. 3, March- 2018, pg. 6-22 **ISSN: 2321-8363**
Impact Factor: 5.515

In this part the performance analysis of proposed Intersect Node Selection Based Clustering Algorithm (INSBCA) with existing Weight Based Adaptive Clustering Algorithm (WBACA). In Fig. 1 the proposed INSBCA algorithm provides better performance compare to existing WBACA and also increasing throughput with transmission range is increased. In Fig. 2 the proposed INSBCA algorithm provides better performance compare to existing WBACA and also increasing throughput with mobility range is increased.

Routing Overhead: It is defined as the ratio of the number of control packets transmitted to the number of the data packets delivered.

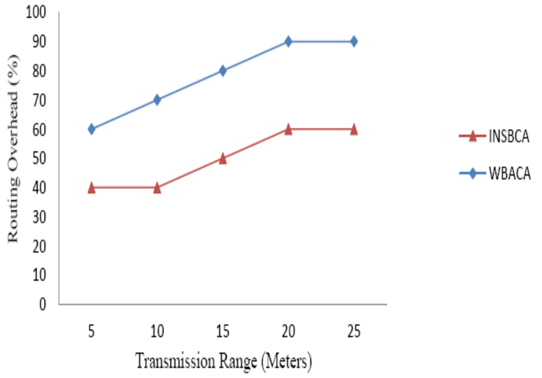

Figure 3: Routing Overhead Vs. Transmission Range.

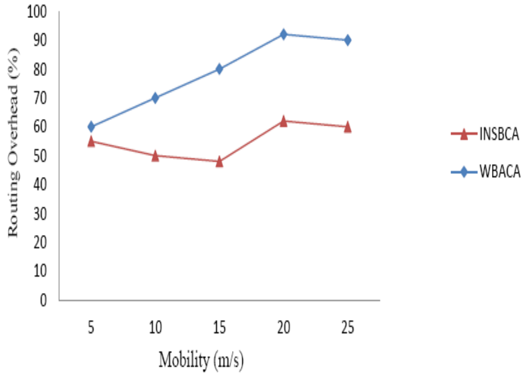

Figure 4: Routing Overhead Vs. Mobility.

In this part the performance analysis of proposed Intersect Node Selection Based Clustering Algorithm (INSBCA) with existing Weight Based Adaptive Clustering Algorithm (WBACA). In Fig. 3 the proposed INSBCA algorithm provides better performance compare to existing WBACA and also reducing routing overhead with transmission range is increased. In Fig. 4 the proposed INSBCA algorithm provides better performance compare to existing WBACA and also reducing routing overhead with mobility range is increased.

End-to-End Delay: End-to-end delay indicates the time lapse between the source and destination nodes in the network. On increasing the mobility of the nodes, the delay increases due to reconfiguration of the network topology. The end-to-end delay also increases due to increase in the number of nodes due to more number of hops.

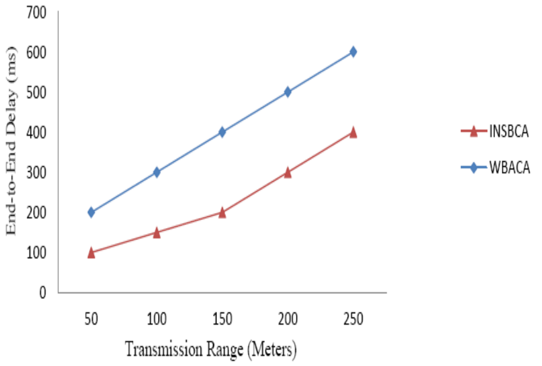

Figure 5: End-to-End Delay Vs. Transmission Range.

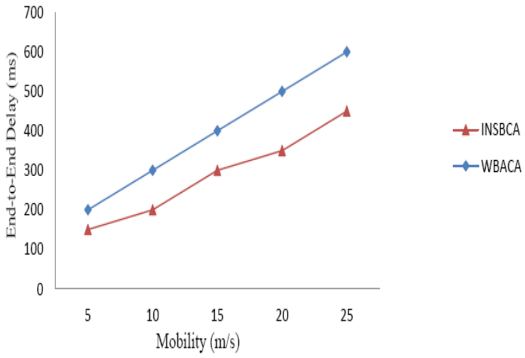

Figure 6: End-to-End Delay Vs. Mobility.

In this part, the performance analysis of proposed Intersect Node Selection Based Clustering Algorithm (INSBCA) with existing Weight Based Adaptive Clustering Algorithm (WBACA) has been made. In Fig. 5 the proposed INSBCA algorithm provides better performance compare to existing WBACA and also reducing end-to-end delay with transmission range is increased. In Fig. 6 the proposed INSBCA algorithm provides better performance compare to existing WBACA and also reducing end-to-end delay with mobility increased.

6. Conclusions

MANET technology is used for communications during emergency situations like disaster management and military deployment, which do not have any fixed infrastructure. This has drawn much attention for research, due to its ad hoc nature. The infrastructure based cellular architecture sets up base stations to support the node mobility. Thus mapping the concepts of base stations into MANET could meet its challenges like limited battery power, scalability, available bandwidth, etc. This leads to the design of logical clusters, where the clusterheads in every cluster play the role of base station. The clusterheads also form the virtual backbone for routing the packets in the network. In this paper, proposed new routing algorithm named Intersect Node Selection Based Clustering Algorithm (INSBCA) for MANET in three different scenarios and its result compare with existing clustering algorithm named Weight Based Adaptive Clustering Algorithm (WBACA). This proposed INSBCA provide better performance compares to the existing algorithm and also increasing throughput, reducing end-to-end delay and routing overhead with transmission range and mobility is increased.

REFERENCES

[1] H.J.Cha, J.M. Kim and H.B. Ryou, "A study on the clustering scheme for node mobility in mobile Ad-hoc network", In *Advanced in Computer Science and its Applications*, Springer Berlin Heidelberg, Vol(279), Iss(1), PP:1365–1369, 2014.

[2] Mohd. Junedud Haque, Mohd Muntjir and Hussain Abu Sorrah, "A Comparative Survey of computation of cluster head in manet", *International Journal Of Computer Applications*, Vol(118), Iss(3), Pp: 6–9, 2015.

[3] Amit Gupta, "Weighted Clustering Algorithms in MANET: A Survey", *International Journal of Latest Trends in Engineering and Technology* (IJLTET),Vol(5), Iss(3) , PP: 316-323, May 2015.

[4] Faraz Ahsan and Akhtab Hussain Khalid Hussain, Abdul Hanan Abdullah, Khalid M. Awan, “Cluster Head Selection Schemes for WSN and MANET: A Survey”, *World Applied Sciences Journal*, Vol (23), Iss (5), PP: 611–620, 2013.

[5] K. Prabu and A. Subramani, “Performance comparison of routing protocol in MANET”, *Int. J. of Adv research in Com. Sci. and Soft Engg* (IJARCSSE), Vol(2), Iss(9), PP:388–392, 2012.

[6] Y.Y. Su, S.F. Hwang, and C.R. Dow, “An Efficient Cluster-Based Routing Algorithm in Ad hoc Networks with Unidirectional Links”, *Journal of Information Science and Engineering*, Vol (24), Pp: 1409–1428, 2008.

[7] Renu Popli, Dr.Kannwal Garg and Sahil Batra, “A Hierarchical classification of various clustering schemes for MANET”, *IJIRAE*, Vol (2), Iss (7), Pp: 110-113, July 2015.

[8] Mrunal Gavhale, Pranav D. Sarat, “Survey on Algorithms for Efficient Cluster Formation and Cluster head selection in MANET”, *International Conference on Information Security and Privacy*, Vol (78), pp: 477– 482, 2015.

[9] Piyalikar, Pritam Karb, and Mrinal Kanti Deb Barma,” Forecast Weighted Clustering in MANET”, *Procedia Computer Science* 89, PP 253–260, 2016.

[10] S.K.B. Rathika and J. Bhavithra. “An efficient fault tolerance quality of service in wireless networks using weighted clustering algorithm”, *Bonfring International Journal of Research in Communication Engineering*, Vol (2), Iss (1), pp: 1–6, 2012.

[11] Kirti Tiwari, K.K Joshi and Neelam Joshi, “An Enhanced Weighted Clustering Algorithm for MANET”, *International Journal for Scientific Research and Development*, Vol (3), Iss(5), pp: 2321-0613, 2015.

[12] K. Prabu, and A. Subramani, “Energy efficient routing in MANET through edge node selection using ESPR algorithm”, *Int. J. Mobile Network Design and Innovation*, Vol (5), Iss (3), pp: 166–175, 2014.

[13] Ghaidaa Muttasher Abdulsaheb, et al., “Improving Ad hoc Network Performance by Using an Efficient Cluster Based Routing Algorithm”, *Indian Journal of Science and Technology*, Vol(8), Iss (30), PP: 1–8, Nov 2015.

[14] Supreet Kaur, Varsha Kumari, “Efficient Clustering with proposed Load balancing Technique for MANET”, *International Journal of computer Applications*, Vol(111), Iss(13), PP: 21–26, Feb 2015.

[15] Maheswari and G. Radhamani, “Clustering Schemes for Mobile Adhoc Networks: A Review”, *International Conference on Computer Communication and Informatics* (ICCCI -2012), January 10–12, Coimbatore, INDIA, 2012.

[16] Rekha M and Subramani A. “Recent Technical Reviews, Issues and Challenges in Mobile Dense Nodes”, *Int J Recent Sci Res*, Vol: 9, Iss: 2, PP:23915–23918, 2018.