International Journal of Applied Engineering Research ISSN 0973-4562 Volume 14, Number 11 (2019) pp. 2757-2772

Distributed Admission Control using Fast Adaptive Neural Network Classifier to assure Quality of Service in MANET

A.Ponraj[1*], K.Kathiravan[2]
[1]*Assistant Professor, Easwari Engineering College, Chennai, Tamil Nadu, India.*
[2]*Professor and Dean, Easwari Engineering College, Chennai, Tamil Nadu, India.*

Abstract:
Ad-hoc wireless networks, boosted by technology advancements are emerging as a key platform for distributed time critical applications. As Mobile Ad-hoc networks (MANET) exhibits high variability in network topology, it becomes a hard problem to guarantee the user's demand for higher and more predictable Quality of Service (QoS) provisioning for multimedia real-time applications. Though several ongoing endeavors have been proposed, they are often designed considering an idealized lower layer models and not for dealing with practical phenomena such as different applications (interference-sensitive, delay-sensitive), different network sizes with link-quality fluctuations or shadow fading environments. In this paper a "Distributed Admission Control using Fast Adaptive Neural Network Classifier (DAC-FANNC)" model that evaluates new solutions for resource management by optimizing applications admission utility is proposed. The proposed model makes use of virtual cluster organization by electing Intelligent Admission Control Agent (IACA) nodes that are distributed across the network for localizing task management among the cluster members. Based on the change in MANET's topology, the IACA proactively predicts, adapts and admits traffic as per the admission control decision made by accurately estimating and classifying QoS parameters such as band width, link expiration time, etc using FANNC. Moreover, the IACA node discovers reliable paths and permits high priority traffic through backbone links increasing throughput and reducing overall delay during data communication. Experimental analysis proves the proposed model enhances the reliability of assured throughput services by minimizing control signaling and improving application quality even in the face of mobility, shadowing and varying load conditions for higher user satisfaction.

Keywords: MANET, virtual cluster, fast adaptive neural network classifier, distributed admission control, intelligent admission control agent, bandwidth, link expiration.

1. INTRODUCTION

The explosive growth of internet and the increasing use of multimedia applications have raised the users' demand for higher and more predictable quality of service. Significant advances in the area of wireless networks has enabled enhancement in processing power and component miniaturization. Rapid deployment and mobility have opened new opportunities for distributed time critical applications such as military [1], civilian search and rescue missions. Applications that run in these environments may share processing and communications resources concurrently while have strict demands on Quality of Service (QoS) [2]. With the increase in offered services, efficient management of available resources becomes decisive in providing the network stability and QoS provisioning. Resource allocation [3] plays a vital role in wireless environment when the resources are in scarce. With diversified traffic operating at different bandwidth requirements [4], the key challenge in the prevailing scenario is to distribute the available channel capacity among multiple traffics to assure QoS.

Despite the efforts made in the last few years to develop quality of service mechanisms, targeting mobile ad-hoc networks and its wide range of applications, the topological change severely restrict the network's ability to provide the required performance. Clustering MANET nodes has obvious advantages with respect to overall network management. This result in reducing latency and also provides more capacity across the network for traffic with longer paths in terms of hops. Ideally, it helps in managing local traffic by keeping it local (within a given cluster). Cluster-based mechanism [5]can also make highly dynamic topology appear less dynamic and larger network appear much smaller. This fact makes researchers to focus on partitioning the multi-hop network into clusters. Introducing clustering in MANET brings in the following benefits: i) it helps to minimize within bound the generation and propagation of routing information. ii) whenever an event occurs, it requires only hierarchical topology specific localized update ie., it makes it possible and sufficient for the mobile nodes within cluster to perform localized update, iii) it facilitates the spatial reuse of resources in a multi-hop environment to increase system capacity, and iv) clustering makes mobility management easy. The main reason for adopting clustering technique in DAC_FANNC is to realize a scalable routing mechanism that is efficient in an environment that is subject to dynamically changing network conditions.

Furthermore, the impact of MANET topology is hard to predict and manage due to its dynamic nature. Though many existing research works manages admission control by applying computational intelligence techniques [6] such as genetic algorithm, fuzzy logic, and multi-criteria decision making techniques for higher users' satisfaction, it requires an efficient and intelligent call admission control which can take care of this contradicting MANET environment to optimize the resource utilization [7]. This paper exploits the benefits of Fast Adaptive Neural Network Classifier

(FANNC), an emerging and promising technique that exhibits both adaptive and resonance theory to achieve fast learning and high predictive accuracy for resource management in MANET. The incremental learning ability and adaptive characteristic of FANNC makes it fit for real-time environments. The proposed "Distributed Admission Control using Fast Adaptive Neural Network Classifier (DAC-FANNC)" focuses on designing a new QoS aware admission control protocol for supporting delay sensitive critical applications. Aim is to,

- Optimize resource utilization and application quality through virtual cluster organization by localizing admission control among the set of cluster members.
- Enforce an incremental learning capability and adaptive characteristic using FANNC to best fit to dynamic changes.
- Predict, adapt and admit diversified traffic for efficient management of available resources.
- Perform admission control for high priority traffic via reliable backbone links.

The proposed model adapts an intelligent and powerful tool of FANNC which has the property to learn from experience. The proof of concept to simulate the proposed DAC_FANNC methodology was developed using MATLAB. Experimental analysis proves that the proposed strategy exhibits better efficiency in handling incoming traffic keeping the resource utilization at an optimal level. The proposed approach would be useful for implementation in military applications where natural clusters of MANET nodes may already exist, but requires increased management efficiency.

The rest of the work is organized as follows: in section 2 related works is discussed; in section 3 detailed illustration of the proposed DAC_FANNC system is elaborated. Section 4 presents an evaluation of the system in terms of its performance. In Section 5, the paper concludes along with a discussion on future directions.

2. RELATED WORKS

Mobile ad-hoc wireless networks are characterized by link volatility, dynamic topology and small factor, therefore resource management systems specifically designed for wired networks cannot be plugged in directly. New architectures and protocols are required to accommodate the wireless dynamic environment. A key component of any resource management system is the network RM. Guaranteeing delay and Mobile ad-hoc wireless networks are characterized by link volatility, dynamic topology and small factor, therefore resource management systems specifically designed for wired networks cannot be plugged in directly. New architectures and protocols are required to accommodate the wireless dynamic environment. A key component of any resource management system is the network RM. Guaranteeing delay and Communication over shared medium in a multi-hop wireless network [8] demands a different perspective on network QoS AC management techniques [9-11]. In practice, AC models for multi-hop networks pose more challenges due to its distributed and lack of centralized control mechanism. In this section, we will discuss some of the related Admission control models coupled with routing schemes such as Perceptive Admission Control (**PAC**), Contention- Aware Admission Control (**CACP**), IN-band signaling and the temporally Ordered Routing Algorithm (**INORA**) and Adaptive Admission Control (**AAC**).

Perceptive Admission Control (PAC) [12] performs passive monitoring by using channel busy time slot to monitor the wireless channels. Using channel busy time the available capacity is estimated and the admission control decision is adapted by the current node and its neighbors. This enables high network utilization while preventing congestion. Though this scheme has the advantage of being utilized in QoS aware strategy, the drawback of this approach is, it does not consider intra-flow interference when making admission control decisions.

To provide AC decision for single and multiple channel ad-hoc networks, a Contention-Aware Admission Control (CACP) [13] model that uses the knowledge of both local resources and the effect of admitting flows to neighboring nodes is proposed. CACP characterizes the contention in the network. Each node in CACP makes AC decision by making use of its c-neighbors (nodes in carrier sensing range) available bandwidth. On-demand querying packets are crucial for effective AC. Moreover, loss of packets results in inaccurate and unreliable AC in this scheme.

A unified signaling and routing admission control mechanism called INORA (IN-band signaling and the temporally Ordered Routing Algorithm) [14] employs the combination of TORA [15] and INSIGNIA [16] protocols. In this scheme the routing information is assumed to be already discovered by TORA and maintained in the destination node. The data packets are automatically admitted whenever a flow request arrives as a soft-state reservations are set using INSIGNA components. The directed graph derived by TORA is followed by the data packets during the admission process, while if an intermediate node finds that it has insufficient resource to accommodate the request flow, it attempts to route the session via different downstream nodes. Whereas, if almost all intermediate nodes have enough resource to accommodate the request flow, then reservations are made to choose node's that support at least the session's minimum required throughput along the path. Though INORA cooperatively supports the session through multiple paths, it employs this approach only under a simplified interference setup.

An Adaptive Admission Control (AAC) model [17] to deal with issues related to QoS provisioning in MANET. In AAC, the Admission control mechanism is coupled with QoS-AODV-style route discovery. It provides an low-cost signaling technique to accurately retrieve available bandwidth and implements an contention count calculation algorithm to adapt to path's roughness. The usable bandwidth information along with one hop neighbor data is spread through Hello message to only one hop. It accurately estimates the residual capacity at nodes during traffic admission, works fine in moderate traffic load and best suits for small-scale networks.

Most of these schemes can be used efficiently for a small to medium network sizes but when the network becomes large, efficient use of resources in such situation should be carefully considered. Moreover, our approach tend to reduce the amount of information stored in intermediate nodes and make the admission control process lighter and flexible by implementing a virtual cluster organization approach, which is discussed in detail in the following section.

3. SYSTEM DESCRIPTION

The System architecture of the proposed network consists of randomly distributed mobile nodes with varying transmission power and communication capability connected among each other. To prevent manual interventions and cope-up with dynamic change in environment, the system is *virtually clustered* to self-organize and re-configure itself to any variations.

Let us assume the network consists of N nodes with each node can either act as a source or destination or relay or a head node. Let the radio transmission range of each node be referred by T_R. The distance between node i and j be denoted by d_{ij} provided if $d_{ij} \leq T_R$, then the nodes i and j are said to be neighbors as they can directly communicate with each other. Let R bits/seconds be the transmission rate. Let the average packet arrival rate at a node is λ packets/sec. Then, the packets generated by the node as well as packets arrived from neighbors is given by,

$$\lambda = n \times \lambda_M \tag{1}$$

where, λ_M is the number of packets generated by node M and n indicates number of neighbor nodes including itself. Further, the size of each packet is assumed to be constant. Let each node in the network has a total of B_j basic bandwidth units (bbu). The bandwidth of a session indicates the number of bbu that is adequate for guaranteeing the desired QoS for the session. It is assumed that packet-level QoS is assured by allocating at least the minimum bandwidth.

During the network deployment stage virtual clusters (VC) are formed and the nodes are organized within the clusters.

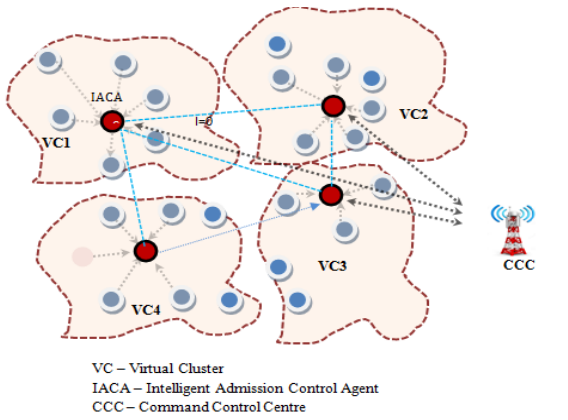

Fig.1. Network Model of DAC_FANNC System

Each VC comprises of N_{opt} nodes provided, $N_{min} \leq N_{opt} \geq N_{max}$, where N_{min} and N_{max} indicates minimum and maximum number of nodes in each clusters. Weighted score for each node is derived among the cluster members and the node with high weighted score is elected as the head node which is also referred to as *Intelligent Admission Control Agent (IACA)*. The role of distributed IACA is to perform localized admission control among the members thereby optimize resource utilization and application utility. The Network model of the proposed DAC_FANNC is shown in Fig 1.

Unlike other AC models where AC process is executed in each node to perform session admission activity, the proposed DAC_FANNC model enables and executes AC process only in the IACA node. Fast Adaptive Neural Network classification plays a vital role in IACA's functionality as it not only classifies diversified traffic efficiently using available resources, but also facilitates incremental learning of different characteristics making the model to self-adapt and best fit to dynamic changes. Nomenclature used in DAC_FANNC is summarized in Table 1.

Table 1. Summary of Notations used in DAC_FANNC System

Notation	Description
VC_{Node}	Virtual Cluster Node
VC_{Head}	Virtual Cluster Head
T_R	Transmission Range
R	Transmission Rate
λ_M	Number of packets generated by node M
bbu	Basic bandwidth units
N_{opt}	Optimal number of nodes
N_{min}	Minimum number of nodes
N_{max}	Maximum number of nodes

The Distributed Admission Control using Fast Adaptive Neural Network model comprises of the following key components:

Virtual Cluster Node (VC_{Node}): These nodes are the members of virtual cluster and are referred to as *Virtual Cluster Node*. It maintains Local Aware Table (LAT) which consists of neighbor nodes and head node's identity. The VC_{Node} directly performs data communication for medium/low profiled requests and directs high profiled requests to its virtual head node which is also called as *Intelligent Admission Control Agent.*

Virtual Cluster Head (VC_{Head}) or Intelligent Admission Control Agent (IACA): These nodes are head of the virtual cluster and are referred as *Virtual Cluster Head*. It is also called as *Intelligent Admission Control Agent.* High profiled traffic request from VC_{Node} are sent to IACA which are further classified and admitted as per the network resource availability to the destination. Node with high weighted score is selected as the head node of the cluster group.

Centralized Control Centre (CCC): The centralized control center coordinates activities performed across the network and initiates cluster formation activity based on the changes in network topology. Using the information about the overall structure and nature of the network relevant activities across the network is carried out by the proposed approach using the CCC.

The proposed DAC_FANNC System consists of the following phases:

Phase I – Virtual Cluster Formation and Selection of Intelligent Admission Control Agent

Phase II - Distributed Admission Control using Fast Adaptive Neural Network Classifier (DAC-FANNC) Mechanism

1. Phase 1 – Virtual Cluster Formation and Selection of Intelligent Admission Control Agent

Objective and purpose of clustering is two-fold: i) to create a hierarchical network, and ii) to select a dominating set of nodes called virtual cluster heads. Aim is to construct stable clusters by considering node-mobility as its primary criterion. In MANET, stable cluster formation is very much essential for better QoS as the performance metrics such as throughput and delay are tightly coupled with each other based on the cluster organization and fequency of cluster reorganization.

Initially, the whole network is divided into virtual groups called virtual clusters using virtual cluster formation (VCF) process. The criteria considered for virtual cluster formation are, i) nodes in each cluster should be within the transmission range. ii) each cluster is restricted to have an optimal number of member nodes say, N_{opt} depending on the converage area such that $N_{min} \geq N_{opt} \leq N_{max}$, where N_{opt} indicates the optimal number of nodes within a cluster, N_{max} and N_{min} indicates maximum limit and minimum limit of nodes that should reside in each cluster. iii) Anonymous node formation is avoided by preventing overlapping. During this phase, nodes in the network are selected in random fashion to intiate VCF. Virtual clusters are formed in such a way that the network is virtually cluster connected. VCF among the nodes in the network is initiated through System Information Message (SIM) broadcast. The steps referred below describes the VC formation and Intelligent Admission Control Agent selection process:

Step 1: Neighbor node discovery is the first step initiated during virtual cluster formation phase. Nodes in the network are randomly selected to broadcast ‘Hello’ message with SIM such as,

Hello [$\mathbf{N}_{\mathbf{id}}^{i}$, $\mathbf{N}_{\mathbf{cp}}^{i}$, $\mathbf{N}_{\mathrm{mo}}^{i}$, $\mathbf{N}_{\mathbf{mc}}^{i}$]

The SIM consists of identity of the node ($\mathbf{N}_{\mathbf{id}}^{i}$), computing capability of the node ($\mathbf{N}_{\mathbf{cp}}^{i}$), mobility of the node ($\mathbf{N}_{\mathrm{mo}}^{i}$), memory capacity of the node ($\mathbf{N}_{\mathbf{mc}}^{i}$) etc.

Step 2: Neighbor nodes within the transmission range of the sender receives the ‘Hello’ message registers the sender’s SIM information in its LAT. LAT is the Local Aware Table maintaned by all the nodes to store information regading its neighbors.

Step 3: After recording the SIM information of the sender in LAT, the neighbor node responds back to the sender by sending an acknowedgement message ‘Ack’. Upon receiving the ‘Ack’ message,the sender records the SIM information of its neighbors in its LAT.

Step 4: Using the information recorded in LAT, every node calculates its as well as its neighbor’s weighted score using the parameters such as $\mathrm{N}_{\mathrm{id}}^{i}$, $\mathrm{N}_{\mathrm{cp}}^{i}$, $\mathrm{N}_{\mathrm{mo}}^{i}$, $\mathrm{N}_{\mathrm{mc}}^{i}$. Weighted score of each node is calculated and the node which satisfies the criteria such as low mobility, greater computation capability and high memory capacity is given the higher score.

Let the subset of nodes that satisfy the memory and storage capacity requirement be $(\mathrm{u}_1^1, \mathrm{u}_2^1, \dots \mathrm{u}_{\mathrm{ms}}^{\mathrm{k}})$. Similarly, the subset $(\mathrm{u}_1^1, \mathrm{u}_2^1, \dots \mathrm{u}_{\mathrm{mt}}^{\mathrm{k}})$ indicates the number of nodes satisfying both mobility and transmission power. The virtual cluster head or the Intelligent Admission Control Agent selection considers the following parameter as referred in Table 2,

Table 2. SIM data of nodes in LAT

Node id ($\mathbf{N}_{\mathbf{id}}^{i}$)	Mobility ($\mathbf{M}_{\mathbf{n}}^{i}$)	Transmission power ($\mathbf{Tp}_{\mathbf{n}}^{i}$)	Computing Capability ($\mathbf{C}_{\mathbf{n}}^{i}$)	Storage Capacity ($\mathbf{S}_{\mathbf{n}}^{i}$)
1	$\mathrm{M}_{\mathrm{n}}^{1}$	$\mathrm{Tp}_{\mathrm{n}}^{1}$	$\mathrm{C}_{\mathrm{n}}^{1}$	$\mathrm{SC}_{\mathrm{n}}^{1}$
2	$\mathrm{M}_{\mathrm{n}}^{2}$	$\mathrm{Tp}_{\mathrm{n}}^{2}$	$\mathrm{C}_{\mathrm{n}}^{2}$	$\mathrm{S}_{\mathrm{n}}^{2}$
N	$\mathrm{M}_{\mathrm{n}}^{\mathrm{n}}$	$\mathrm{Tp}_{\mathrm{n}}^{\mathrm{n}}$	$\mathrm{C}_{\mathrm{n}}^{\mathrm{n}}$	$\mathrm{S}_{\mathrm{n}}^{\mathrm{n}}$

Using the set of selection contraints, the node with high transmission power, low mobility, high computation capability and high storage capacity is elected as the head or the IACA node. ie,

$$\min_{x \in M_x} \min_{x \in Tp_x} \sum_{1=1}^{n} (\max M_{ij} :\{K_i y_i \geq b_i - Gi_x \geq 0\}) + \sum_{1=1}^{n} (\max Tp_{ij} :\{L_i y_i \geq b_i - Hi_x \geq 0\}) \quad (1)$$

$$\min_{x \in A_x} \min_{x \in B_x} \sum_{1=1}^{n} (\max\ C_{ij} :\{Q_i y_i \geq b_i - Pi_x \geq 0\}) + \sum_{1=1}^{n} (\max B_{ij} :\{SC_i y_i \geq b_i - Ri_x \geq 0\}) \quad (2)$$

subject to the following criteria’s:

$$Gi_x + K_i y_i \geq b_i \forall ix \in M_x y_i \geq 0\ \forall i,\ Hi_x + L_i y_i \geq b_i\ \forall ix \in Tp_x y_i \geq 0 \forall i,\ Pi_x + Q_i y_i \geq b_i \forall ix \in C_x y_i \geq 0 \forall i,\ Ri_x + S_i y_i \geq Ri_x + S_i y_i \geq b_i\ \ \forall ix \in SC_x y_i \geq 0\ \ \forall i$$

where ‘i’ refers the number of nodes in the network, b_i is the number of virtual cluster head nodes (or IACA), $K_i y_i$, $L_i y_i$, $S_i y_i$ and $Q_i y_i$ refers to set of nodes with optimal mobility, higher transmission power, higher storage capacity, and optimal storage capacity. Similarly, Gi_x indicates set of nodes that satisfy mobility criteria, Pi_x refers to set of nodes under higher computing memory, M_x indicates nodes with low mobility and uniform trajectory, Tp_x is higher transmission power, C_x is higher computation power of node, Ri_x is set of nodes under higher storage capacity, SC_x denotes storage capacity of nodes in the network.

Thus, max $z = \sum_{i=1}^{l} \beta_i$ subject to $\beta_i \geq u_{mt}^{k} (b_i - Gi_x) \forall i \; \forall$ by comparing Tp_n^i, A_n^i, B_n^i, ie the node's transmission power, memory and storage with the limit, subject to $\beta_i \geq u_{mt}^{k} (b_i - Hi_x) \forall i \forall k$ and 'l' contains the list of selected IACAs after applying the limit. The subset containing $(u_1^1, u_2^1, \dots u_{ms}^k)$ is compared with limit, max $z = \sum_{i=1}^{l} \beta_i$, subject to $\beta_i \geq u_{ms}^{k} (b_i - Pi_x) \quad \forall i \quad \forall k$ and $\beta_i \geq u_{ms}^{k} (b_i - Ri_x) \quad \forall i \quad \forall k$. If the optimal solution satisfies all the constraints, then $z_i \in \{0,1\}$, $i = 1,2, \dots n$ is the obtained solution. ie.

$$\text{Min } \sum_{i=1}^{n} M_n^i + \sum_{i=1}^{n} Tp_n^i + \sum_{i=1}^{n} C_n^i + \sum_{i=1}^{n} S_n^i$$

Step 5: The weighted score of each node is compared with its neighbors, among the set of nodes, the one with the highest score is is declared as the virtual cluster head or the IACA node.

Step 6: The Node selected as virtual head (also called as Intelligent Admission Control Agent) propagates virtual group formation "*VGF_Msg*" message to its neighbors. Neighbors receives the "*VGF_Msg*" message, responds back by sending "*VGF_Confirm*" message and become part of the virtual cluster group. Once the neighbor member node becomes part of the cluster, any further virtual group formation messages or re-current group formation request messages are discarded by those nodes.

The virtual cluster formation process involved in DAC_FANNC mechansim is depicted in Algorithm 1.

Algorithm 1: Virtual Cluster Formation in DAC_FANNC

//Node I broadcasts SIM

$N_I \rightarrow N_I\_SIM(N_{Id}, N_{Mb}, N_{Tp}, N_{Cc}, N_{Sc})$;

// Neighbor Node J receives SIM of N_I

$N_J \leftarrow N_I\_SIM(N_{Id}, N_{Mb}, N_{Tp}, N_{Cc}, N_{Sc})$

If (N_J (Hello_Msg_nodeI) == 'T') then

// Node J receives SIM of Node I successfully

$LAT_J \leftarrow$ fetch_data($N_{Id}, N_{Mb}, N_{Tp}, N_{Cc}, N_{Sc}$);
//fetch the data of Node I and store in LAT of Node J

$N_J \rightarrow$ Ack_Msg;

//Node J responds with "Ack_Hello" message

If (N_I (AckMsg_nodeJ) == 'T') then
//Node I receives "Ack_Hello" message of Node J

$LAT_I \leftarrow$ fetch_data($N_{Id}, N_{Mb}, N_{Tp}, N_{Cc}, N_{Sc}$);

// fetch the data of N_J and store in LAT of N_I

end;

end;

/* Register SIM of all neighbors nodes. Find weighted score.*/

$Score_{data}$ = calculate_score(LAT_I);
$Score_{high}$ = get_highest_score($Score_{data}$);

// Scores are compared to find the highest score

$Node_i$ = get_node($score_{high}$);

// Node with highest score is identified as the IACA node.

$VGH_{node} = Node_i$;

$VGH_{node} \rightarrow$ VGF_Msg;

// IACA propagates Virtual Group Formation msg to neighbors

$Node_{neighbor} \leftarrow$ VGH_Msg ;

//Neighbors receives VGF message and becomes part of cluster

$Node_{neighbor} \rightarrow$ VGF_Confirm;

// Neighbors send confirmation message to VGH_{node} and the Virtual Group Head node the IACA is identified.

Step 7: The node elected as IACA or the virual head records the SIM information of its member nodes in its LAT – the table that maintains the member node list. LAT maintains up-to-date view of the registered members identity. Thus, the neighbor members becomes part of the virtual cluster. The node selected as the virtual group head registers itself as the IACA node of the cluster.

The process of virtual cluster formation is initiated after 't_i' time slot, ie., after 't_i' time slot, if the current IACA or the head node is found to be inappropriate to act as the virtual head node, it sends "*VGF_Msg*" message to the node 'N_{nxtopt}' (next optimal highest weighted score node) in the group to become the IACA and waits for a fixed duration 'δ'of time to receive the response from the node 'N_{nxtopt}'. If the 'N_{nxtopt}' node or any other node in the cluster fails to respond to the "*VGF_Msg*" message, then the Current IACA node remains to be the head node. Similarly, virtual groups across the network performs the IACA selection process as referred in Algorithm 2.

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 14, Number 11 (2019) pp. 2757-2772
© Research India Publications. http://www.ripublication.com

Algorithm 2: The Intelligent Admission Control Agent selection Process in DAC_FANNC

Let $\mathrm{M}_{\mathrm{n}}^{i}$, $\mathrm{Tp}_{\mathrm{n}}^{i}$, $\mathrm{Sc}_{\mathrm{n}}^{i}$, $\mathrm{Mem}_{\mathrm{n}}^{i}$, $\mathrm{V}_{\mathrm{count}} = 0$, j=1;

Step 1: for set of 'n' nodes

if($(\mathrm{N}_{\mathrm{VCH}}^{i}\ (\mathrm{Tp}_{\mathrm{VCH}}^{\mathrm{n}}) \geq \mathrm{Tp}_{\delta})$ &&$(\mathrm{N}_{\mathrm{VCH}}^{i}(\mathrm{M}_{\mathrm{VCH}}^{\mathrm{n}}) \leq \mathrm{M}_{\delta}$)) ||

$((\mathrm{N}_{\mathrm{VCH}}^{i}(\mathrm{Mem}_{\mathrm{VCH}}^{\mathrm{n}}) \geq \mathrm{Mem}_{\delta})$&&$(\mathrm{N}_{\mathrm{VCH}}^{i}(\mathrm{Sc}_{\mathrm{VCH}}^{\mathrm{n}}) \geq \mathrm{Sc}_{\delta})$)then

$\mathrm{Satisfy}_{\mathrm{VCH}}$ = store_data ($\mathrm{M}_{\mathrm{VCH}}^{\mathrm{n}}$, $\mathrm{Tp}_{\mathrm{VCH}}^{\mathrm{n}}$,$\mathrm{Sc}_{\mathrm{VCH}}^{\mathrm{n}}$,$\mathrm{Mem}_{\mathrm{VCH}}^{\mathrm{n}}$);

$\mathrm{VCH}_{\mathrm{ct}} = \mathrm{VCH}_{\mathrm{ct}} + 1$;

end;

end;

Step 2: if $\mathrm{VCH}_{\mathrm{ct}} > \mathrm{P}_{\mathrm{VCH}}^{\mathrm{i}}$

//$\mathrm{Satisfy}_{\mathrm{IACA}}$ is number of nodes satisfying the norms

for i=1 to N_$\mathrm{Satisfy}_{\mathrm{IACA}}$

$\mathrm{optimal}_{\mathrm{count}} = 0$;

if(($\mathrm{Satisfy}_{\mathrm{IACA}}{}_{mob}^{i} \leq \delta_{\mathrm{adj}}$)&&($\mathrm{Satisfy}_{\mathrm{IACA}}{}_{tp}^{i} \geq \delta_{\mathrm{adj}}$))||(($\mathrm{Satisfy}_{\mathrm{IACA}}{}_{sc}^{i} \leq \delta_{\mathrm{adj}}$) &&($\mathrm{Satisfy}_{\mathrm{IACA}}{}_{mem}^{i} \geq \delta_{\mathrm{adj}}$)) then

$\mathrm{O}_{\mathrm{IACA}}$=store_data($\mathrm{Satisfy}_{\mathrm{IACA}}{}_{mob}^{i}$, $\mathrm{Satisfy}_{\mathrm{IACA}}{}_{tp}^{i}$, $\mathrm{Satisfy}_{\mathrm{IACA}}{}_{sc}^{i}$,$\mathrm{Satisfy}_{\mathrm{IACA}}{}_{mem}^{i}$);

$\mathrm{optimal}_{\mathrm{count}} = \mathrm{optimal}_{\mathrm{count}} + 1$;

else if $\mathrm{V}_{\mathrm{count}} < \mathrm{P}_{\mathrm{IACA}}^{i}$

re-adjust threshold limit;

repeat step 1;

end

end

end;

send_IACA ($\mathrm{IACA}_{\mathrm{reg}}$) $\rightarrow$ CCC;

Repeat step (1) and (2) for effective IACA selection.

The frequency for re-electing IACA depends on application.

The IACA registration ($\mathrm{IACA}_{\mathrm{reg}}$) message is sent from Intelligent Admission Control Agent to Centralized Control Center (CCC). CCC receives the $\mathrm{IACA}_{\mathrm{reg}}$ message from the IACA node and updates the IACA registration list ($\mathrm{IACA}_{\mathrm{reg\_list}}$) maintained by the server. The IACA registration list contains the registered list of IACAs selected (along with its registered members) across the network.

2. Phase 1I – Distributed Admission Control using Fast Adaptive Neural Network Classifier Mechanism

To assure Quality of Service (QoS) for real time multimedia applications, MANETs requires QoS aware mechanisms for channel access with acceptable channel conditions, as well as identification of more appropriate forwarding neighbor nodes. Considering the above requirements and depending on the type, intensity of the traffic, and relative mobility patterns of nodes, the proposed DAC-FANNC approach exploits two important functionality,

1. *Localized Service Management* (LSM) and
2. *Distributed Fast Adaptive Admission Control (DFAAC)*

Detailed illustration of the important functionalities performed by the DAC_FANNC is discussed in the following section.

2.1 Localized Service Management:

Unpredictable behavior of channels, time-varying bandwidth, node-mobility in MANET makes it extremely difficult to identify most reliable forwarding neighbor nodes as well as in designing a scalable and efficient routing strategy. As MANET requires routing over multi-hop wireless paths for peer-to-peer communication, route discovery becomes challenging as the multi-hop paths constitute of links whose end points are likely to be moving independently of one another. Employing clustering technique in the proposed approach justifies the effect of *localized service management* and thereby devises a scalable routing strategy that is efficient subject to dynamic change in network conditions making it suitable for large scale mobile ad-hoc networks. The performance of the routing strategy is judged based on its effectiveness, efficiency and route properties such as

- *route optimality* – indicates the shortest-path depending on the type of traffic
- *route latency* – indicates the time taken by the routing mechanism for a given source to obtain the initial routing information for its desired destination
- *route stability* – refers to the random mobility patterns of the node and hence relates to how long the found route will be available.
- *route diversity* – indicates for a given source-destination pair, whether the disjoint paths would be maintained partially or completely for the same flow.

In DAC_FANNC, the IACA registration ($\mathrm{IACA}_{\mathrm{reg}}$) message sent from various IACAs across the network are received and registered by the CCC which stores the complete information of newly elected IACAs (along with its registered members). The following steps are performed for localized service management and legitimate route discovery,

Step 1: After registering the newly elected IACA, the CCC sends back a registration acknowledgement *Reg_Ack*[$\mathrm{IACA}_{\mathrm{reg\_list}}$] response to each IACA node. The *Reg_Ack* message consists of registered list of IACAs elected across the network.

Step 2: Upon receiving the *Reg_Ack* message, the IACAs in each cluster retrieves the $\mathrm{IACA}_{\mathrm{reg\_list}}$ from the *Reg_Ack* message and initiates IACA registration process to update its LAT and Global Aware Table (GAT). The GAT maintains the up-to-date view of the registered IACAs ($\mathrm{IACA}_{\mathrm{reg\_list}}$) across the network at a particular instant of time.

Step 3: Apart from maintaining the LAT, each Intelligent Admission Control Agent also maintains a GAT which contains the latest list of registered IACA node's elected across network (used for legitimate route discovery)

Step 4*:* Upon performing the IACA registration process, the IACA enables a fast adaptive neural network based admission control process to estimate the node's state and to realize service access for higher priority traffic classes through legitimate backbone link.

Step 5*:* Data communication in DAC_FANNC model is performed in two ways based on the service differentiation (data packets are categorized either as high priority or as best effort traffic) as follows:

i) High priority user data offered by members are transmitted with the help of virtual head or IACA nodes via the backbone links.

ii) Best effort data is transmitted between the source and destination through the path discovered between them.

Step 6: Backbone links are established between the head nodes for transmitting high priority data traffic. Let source node (S) has data to be transmitted to a destination (D) node. Prior to data communication, based on the type of data traffic (D_{type}), the following verification is performed, ie., the source being the member of a virtual cluster verifies if D_{type} belongs to *high priority service class*. If so, any of the case scenario referred below is executed:

Case 1 : If the source node "S" in virtual cluster "VC_1" needs to send high priority data traffic to destination node "D" that belongs to same virtual cluster "VC_1" as shown in figure 2, provided the virtual head node of the cluster is "$IACA_1$" then,

- Node "S" sends a probe request (it contains the source and destination nodes identify) to its cluster head "$IACA_1$"
- From the probe request, the virtual head "$IACA_1$" reads the source and destination nodes identity and looks-up its LAT to verify if the destination node id exists in its LAT.

  If both source node "S" id and destination node "D" id exist in $IACA_1$ node's LAT then

  "$IACA_1$" sends admit request "*Adm_Req*" notification message to source "S", indicating data communication to happen directly between

  source "S" and destination "D"

  end;

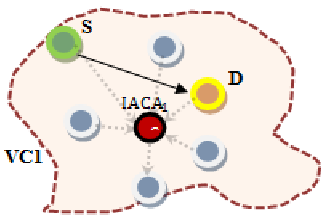

Fig.2. Source node S and Destination node D in same virtual cluster

Case 2 : If the source node "S" in virtual cluster "VC_1" needs to send high priority data traffic to destination node "D" that belongs to another virtual cluster "VC_2" as shown in figure 3, provided the virtual head node of the cluster "VC1" is "$IACA_1$" and the virtual head node of the cluster "VC2" is "$IACA_2$"then,

- Node "S" sends a probe request to its cluster head "$IACA_1$"
- From the probe request, "$IACA_1$" reads the destination nodes identity and looks-up its LAT to verify if the destination node id exists in its LAT. As per the case considered, let the source node id "S" exist in the LAT of VC_1, while destination node id "B" would not. Hence, the head node "$IACA_1$" will now look-up its GAT which contains the global list of cluster head node ids along with its registered member list information.
- Upon verification, $IACA_1$ finds the destination node id "B" to exist in "VC_2" member list, then "$IACA_1$" sends a probe request to "$IACA_2$" of VC_2 to establish a legitimate backbone link between VC_1 and VC_2. The Source node "S" in VC_1 then forwards the high priority data to "$IACA_1$" which admits the data along the backbone link towards the destination node "D" in "VC_2" as per the node status derived using the FANNC process.

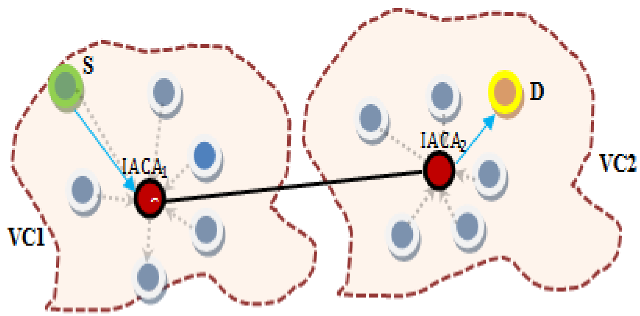

Fig.3. Source node S and Destination node D in different virtual cluster

Step 7: Apart from high priority service class, the *best effort data traffic* is transmitted between the source and the destination via the legitimate path established between them.

Path discovery by IACA: The IACA node exploits a fast adaptive neural network based classification mechanism to adjust to frequently changing traffic conditions, radio propagation and network status. The admission control enabled in IACA node helps to handle high priority service request of member nodes to avoid susceptibility to network failures due to congestion around critical nodes and penalties due to inefficient routing.

Detailed discussion on how the "N_{status}" is derived using the metric such as available bandwidth, LET explained in section 2.2. The steps below illustrate how the path selection using the N_{status} is performed in DAC_FANNC model.

Step1: Let a source have data service to be communicated to a destination, provided the source node is the member of a virtual cluster. Prior to data communication, the member node verifies if the type of the data service (DS_{type}) is $BE_{traffic}$ or $HP_{traffic}$, if the DS_{type} == $HP_{traffic}$, then the member node sends the Req message to its IACA node. The request message, Req[MN_{id}, $IACA_{id}$, DS_{type}, M_{Data} , DS_{id}] consists of member node identity - MN_{id} , intelligent admission control agent identity - $IACA_{id}$, data service type - DS_{type} , measure of data content to be transmitted and destination identity DS_{id}.

Step 2: The IACA node receives the Req message, validates its current *Node Status* N_{status} to evaluate if it has the capability to accommodate the traffic as per the user requirement.

If ($N_{status} < \mu$) then

IACA ignores the request from member;

IACA sends Req_ignore message to member;

end if;

if ($N_{status} >= \mu$) then

IACA verifies its LAT and GAT;

IACA discovers legitimate route;

IACA reserves resources;

IACA accepts and performs data transmission;

end if;

where μ indicates the threshold limit of IACA nodes resource availability based on which the data transmission is admitted or rejected.

Step 3: Let us consider the current status of the node satisfies the criteria such that it has the capability to perform the HP traffic transmission as per the user's QoS requirement. The IACA regulate traffic in a proactive manner by selecting appropriate non-overloaded forwarding neighbor IACA nodes. Each IACA node calculates its node status using the following:

$$N_{status} = (LET_{MI} * BW_{available}) / (d_{MI} + d_{IZ}) \qquad (1)$$

Where N_{status} indicates the current node status derived based on the LET_{MI} - the Link expiration time of node M with respect to its one hop neighbor node I ($I \in N(M)$, where N(M) indicates set of one of neighbor nodes of node M, $BW_{available}$ - available bandwidth and d_{MI} - the distance between the node and its one hop neighbor M, d_{IZ} – the distance between the one hop neighbor node M and the destination node Z.

Step 4: The IACA node in each cluster calculates the N_{status} and shares it with each other through a *IACA_Beacon*[$IACA_{id}$, N_{status}, Timestamp] beacon message, where $IACA_{id}$ – indicates the IACA node identity, N_{status},- indicates the IACA nodes current resource status, Timestamp – indicates the time stamp during which the N_{status} was calculated. The beacon message sent by the IACA is received by the neighbor IACA node and updates its GAT which maintains the list of neighbor IACA node id, N_{status} and timestamp information. At any point in time, by looking-up the GAT, an IACA node can find its neighbor IACA nodes along with their node resource availability.

Step 5: During the path discovery, the IACA node first looks up its GAT to verify,

Case 1: if it has an IACA one hop neighbor which has the destination node in its cluster. If it exists, then, it directly establishes a path to the IACA node and does the data communication.

Case 2 : In case if there does not exist any IACA one hop neighbor with destination node, then, the source IACA node selects set of reliable neighbor IACA node which satisfies its member node's N_{status} requirement. Based on the selection, probe request is sent only to those reliable IACA neighbor nodes. The probe request IACA_PReq (S_{id}, D_{id}, Path, SN_{status}) consists of source member identity, destination node identity, path with first address filled with its own address and the address of IACA's via which the request was forwarded and the source IACA node's status information.

The process continues until the destination node is reached.

Algorithm 3 explains the steps involved in path discovery process.

Algorithm 3: Path Discovery using IACA node.

Initialize N_{status};

While(hasData(Mnode) == 'T')

If (DS_{type} == $HP_{traffic}$) then

Mnode sends Req[MN_{id},$IACA_{id}$, DS_{type},M_{Data} ,DS_{id}] to $IACA_{node}$;

$IACA_{node}$ receives Req[MN_{id},$IACA_{id}$,DS_{type},M_{Data} ,DS_{id}];

$IACAN_{status}$ = calculateNstatus (LET_{MI},$BW_{available}$, d_{MI} , d_{IZ});

If ($IACAN_{status} < \mu$) then

$IACA_{node}$ rejects Req;

$IACA_{node}$ sends Req_{ignore} to Mnode;

end if;

if ($IACAN_{status} >= \mu$) then

checkLATGAT ($IACA_{node}$)

discoverPath();

reserveResource();

performDataCommunication(path);

end if;

```
//path discovery process
function discoverPath()
   SIACAnode sends IACABeacon(SIACAid,Nstatus, Timestamp);
   NIACAnode receives IACABeacon(SIACAid,Nstatus, Timestamp);
   // neighbor IACA receives beacon
   Neighbor_updateGAT(IACAbeacondata());
   lookupGAT();
   if ( S[k1(A)].getid()== Did ) then
      setPAth();
      performDC();
   end if;
  if ( S[k1(A)].getMid() ! Did ) then
      repeat
      S[k1(Rnode )]= select(S[k1(A)]);
      sendProbe (k1(Rnode )) ;
      until Did is found;
end if;
```

The above service management procedure of IACA node ensures traffic congestion among the member nodes making backbone link proactively available for HP traffic class data transmission.

2.2 Distributed Fast Adaptive Neural Network Classifier:

This section explains how DAC_FANNC model performs admission control in IACA node using fast adaptive neuro mechanism depending primarily on three important metrics such as

i) Link Expiration Time

ii) Bandwidth Availability

iii) Mobility

Link Expiration Time (LET): It is the predicted time duration that two neighbor nodes will remain connected. To determine LET, every node embeds an mobility prediction module which is used to collect motion parameters such as radio propagation range, velocity etc., of two neighbor nodes M and I, where (I ε N(M)) to calculate the time duration.

LET calculation in the proposed DAC_FANNC model does not follow any new approach, instead the existing mechanism is slightly modified to use in our approach. Existing mechanisms uses two important approaches for calculating LET, i) it uses received signal strength without using GPS ii) it uses location and mobility information provided by GPS. The prediction accuracy in each of these two approaches is questionable since both uses unrealistic assumptions. Therefore, in our proposed model, we determine LET using each of the above mentioned approaches and try to consider the value, which is the minimum of them in our calculations

The first method considers two-ray ground reflection approximation as radio propagation model. It assumes that between subsequent motion parameter updates, the node-pair maintains constant velocity and does not accelerate. It also assumes the sender power level as a constant such that the received signal strength is indicative of the relative distance.

$$LET_{M1}= \frac{2dv\cos\phi + \sqrt{4d^2v^2cos^2\phi - 4v^2[r^2+d^2]}}{2v^2} \quad (2)$$

The second method assumes that the received signal strength solely depends on the distance of the transmitter, all the nodes in the network to have their clocks synchronized and node-pair concerned maintains constant velocity and does not accelerate.

$$LET_{M2} = \frac{-(ab+cd) + \sqrt{(a^2+c^2)r^2 - (ad-bc)}}{a^2+c^2} \quad (3)$$

Based on our approach, the LET we have considered is,

$LET = \min(LET_{M1}, LET_{M2})$

The measurement of LET considered in our approach serves the purpose in finding the relative mobility patterns of nodes w.r.t other, making it a potential candidate during neighbor node selection.

Bandwidth Estimation at any Node: Measurement based bandwidth estimation process assumes the usage of RTS (Request to Send) and CTS (Clear to Send) frames in-order to minimize exposed terminal and hidden terminal problems. If the time instance t_1 (the time when the packet is ready for transmission) and t_5(the time when the Ack message was received) and the frame size (f_s) of the data is known , the any node can predict the available bandwidth using the formula:

$$\text{Predicted Bandwidth Available } (C_M(t)) = f_s / t_5 - t_1$$
$$= f_s / \Delta t \quad (4)$$

The effect of contention impacts the available bandwidth. In case of high contention, the Δt would increase (ie., $t_5 - t_1$ will be high), ie., increase in Δt in equation (2) will result in lower bandwidth. It is noted that the available bandwidth is measured using successful link layer transmission. The bandwidth available at any node is derived using equation (2), only if the time t_1 and t_5 is known as in figure 4.

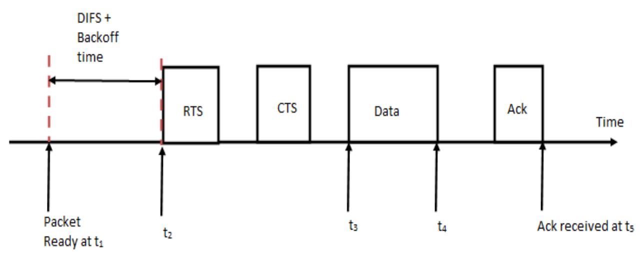

Fig.4. IEEE802.11 Transmission Sequence of Unicast Packet

In real time scenario, it would be difficult for any node to determine the time (t_5) when the Ack packet will be received. Ie, any node can determine the time the data packet is ready for transmission (t_1). Once the RTS is initiated, it would be difficult for any node to determine the exact time when the Ack would be received in return. Hence the value of t_5 is found empirically as follows:

$$t_2 = t_1 + DIFS + \delta_1 \quad (5)$$

$$t_4 = t_2 + t_{RTS} + SIFS + t_{CTS} + SIFS + t_{DATA} \quad (6)$$

$$t_5 = t_4 + SIFS + t_{ACK} + \delta_2 \quad (7)$$

where SIFS indicates short inter-frame space and DIFS indicates distributed inter-frame space. δ_1 is equation (5) is added to take care of an extra time involved to access the channel due to the binary exponential back-off mechanism of the DCF, δ2 is equation (7) is added to take care of an extra time involved if data needs to be retransmitted due to collisions or channel errors. and $t_{RTS,}$ $t_{CTS,}$ $t_{DATA,}$ t_{ACK} are the transmission times for RTS, CTS, DATA and ACK frames respectively.

Using the above equation, it is possible that any node M has the ability to predict the available bandwidth ($BW_{avaiilable}$).

However, based on information related to historical data transmission, (past and present service rate estimation), the average current bandwidth availability of node M at time t_c (ie., $BW_{available}(t_c)$) is referred as follows:

$$BW_{available}(t_c) = (BW_{available}(t_c) * \beta_M + C_M(t_c)) / (\beta_M + 1) \quad (8)$$

Where, β_M indicates number of packets successfully transmitted so far by the node M excluding the current transmission attempt.

To satisfy the QoS guarantees traffic regulation plays a vital role in the network. ie, it becomes necessary to maintain per flow or per-class state information using Admission control and traffic policing mechanism. In case of fixed network, the admission control and traffic policing leads to better performance as the routes taken by packets are not volatile.

While in dynamic networks like MANET, route taken by packets of same flow may tend to vary heavily with time. Therefore, selecting more appropriate non-overloaded forwarding neighbor nodes plays an important role in regulating traffic in a proactive manner in mobile ad-hoc networks. Our proposed FANNC takes into account the link expiration time (LET), currently available bandwidth to a neighbor, and relative locations of the node-pair by applying the fast adaptive neuro-based admission control in the IACA node to select the reliable neighbor node that has the highest value of Ω_{MI} using the following equation:

$$N_{status} = \Omega_{MI} = (LET_{MI} * BW_{available}) / (d_{MI} + d_{IZ}) \quad (9)$$

Let node M has HP packet to be transmitted, and N(M) be the number of neighbor IACA nodes for node M, d_{MI} indicates the distance between the node M and its one hop neighbor I, and d_{IZ} indicates the distance between the neighbor IACA node and the destination node Z. FANNC makes it essential in making use of LET and available bandwidth while selecting neighbor IACA due to the following reasons: i) the nodes may be so busy such that the forwarded packets may face long delays or get dropped ultimately if bandwidth availability is not considered ii) mobility of nodes could cause a situation where the selected node may soon move away from the sender making the packet transmission to fail if LET is not considered.

The DAC_FANNC model generates the N_{status} using the equation (1), which acts as the primary factor in making decision during admission control. This measure of N_{status}, generated by each node indicates the node's current resource availability. It is used to validate the current resource utilization of the node based on traffic, mobility and congestion. The IACA node shares the node status among the neighbors using the beacon packets to make decision during legitimate route discovery.

Fast Adaptive Neural Network Classification mechanism:

Apart from generating the measure of the node status, the distributed fast adaptive neural network based admission control module equipped in IACA node enforces an adaptive incremental learning capability in IACA nodes to best fit to dynamic changes. It incorporates a neural-network based classifier, which has the capability to predict, adapt and admit diversified traffic for efficient management of available resources during data communication.

The fast adaptive neural network classifier is modeled to adapt to new dynamic very quickly and trained to retain knowledge of past dynamic so as to act effectively based on new occurrences. The FANNC module is trained a priori with wide range of historical dynamics, while in parallel is allowed to adapt itself to make up the differences between the historical and real-time dynamics to progressively trained to learn the new dynamics without adversely affecting the old training. The admission control performed using the FANNC module ensures the adaptation process computationally simple and very fast.

The DAC_FANNC model attempts at algorithmic level to optimize the training and learning process of network parameters for efficient implementation. For efficient implementation of fast adaptive neural network classification, the key algorithmic consideration taken into account was the number of neural connections ie., it equals the number of weight parameters and number of multiplication operations. Adopting usual assumptions results in approximately $O(n^2)$ memory and computational complexity.

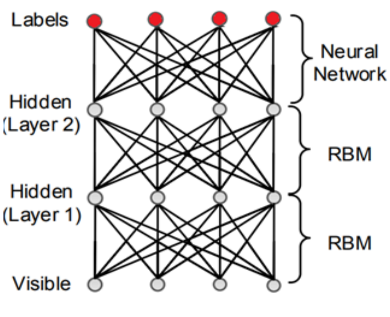

Fig.5. Deep Belief Network (DBN)

By adopting usual assumptions in deep neural networks, as implemented in Deep Belief Network [18], some of the features extracted are negligible for pattern recognition and classification resulting in memory and computational bottleneck as shown in figure 5. Our approach exploits sparsity in neural connections by adaptively reducing or omitting weight parameters associated with negligible features to zero while considers optimization in data representation which is an essential tradeoff for accuracy and cost as shown in figure 6.

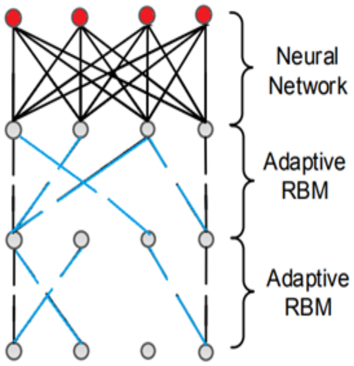

Fig.6. DAC_FANNC Model

As per the research indication, training deep neural networks with limited precision could suffer from significant loss of accuracy, however, in our fast adaptive neural network approach, the sparse weights are naturally separated into the following groups, i) positive weights with large values, ii) negative weights with large absolute values and iii) close-to-zero weights as shown in figure 7. It robustly thresholds and represents the weight parameters using integers without impacting or influencing classification accuracy.

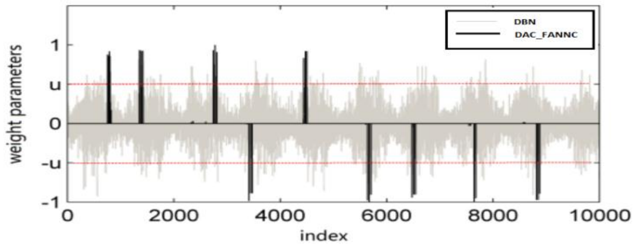

Fig.7. The weight parameter representation of DBN and DAC-FANNC model.

The binary weights can significantly reduce the memory complexity and replace complicated floating-point multiplications by simple logic operations. The DAC_FANNC is designed to extract sparse features as it trims down weight parameters associated with unimportant features. The computational mechanism of FANNC constitutes an important phase called as the "*training phase*".

Training Phase: The parameters are approximated using the training algorithms through high performance computers. Aim of the FANNC training algorithm is to address the challenge involved in memory and computation bottleneck.

The initial stage involves regularization to the sparsely weighted variant to shrink the weights. The mixed norm of a matrix **W** is defined as

$$\| \mathbf{W} \|_M = \Sigma_i \, (\Sigma_j |w_{i,j}|^2)^{1/2} \qquad (9)$$

where, i and j indicates the indices and are treated differently. The shrinking process does not apply evenly to all the rows in the matrix. ie., the stochastic gradient descent process shrinks rows with smaller weights first which shrinks faster. The weights in short rows tend to shrink to zero after finite iterations and then the rows with larger weights are shrinked. Similarly, transposed matrix like $\mathbf{W}^T$ minimizes the weights in short columns to zero. Overall, the FANNC training algorithm extracts important features for classification by shrinking the weight parameters in short rows and short columns to zero and selecting important input and output features from the two matrices **W** and $\mathbf{W}^T$.

For the weight matrix **W,** the output **h** will not be affected by the input feature v_i if the i^{th} row is reduced to zero. On the other hand, for higher level of classification, the probability $p(h_j = 1|\mathbf{v})$ will stay close to a constant $\delta(b_j)$, if the weight parameters w_{ij} associated with h_j is reduced to zero, leading to negligible output feature h_j. The FANNC process being an unsupervised feature selection method assumes that not all features in **v** or **h** are required for higher level of classification, instead by using the regularization method, it aims to shrink and elect the input and output features using the following term:

$$R_s(\mathbf{W}) = \lambda \, (\gamma \, \| \mathbf{W} \|_M + (1-\gamma) \, \| \mathbf{W}^T \|_M) \qquad (10)$$

Where, γ controls the balance between the row sparsity and column sparsity, λ refers to the parameter that controls the sparsity of the weight parameter. The FANNC training algorithm then shrinks the regularization term and then incorporates the following to optimize and set the standard which represents the sum of regularization term and log-likelihood term as follows:

$$\mathrm{argmin}_{\boldsymbol{\theta}} - \Sigma_k \log(\Sigma_h e^{-E(\mathbf{v}(k),\mathbf{h}(k))\mathbf{h})k}) + \lambda(\| \mathbf{W} \|_M + (1-\gamma) \| \mathbf{W}^T \|_M) \qquad (11)$$

The regularization term and derivatives of the log probability w.r.t the parameters are represented as:

$$\partial \log p(\mathbf{v})/\partial w_{ij} = \langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model} + \lambda \partial Rs(\mathbf{W})/\partial w_{ij} \qquad (12)$$

$$\partial \log p(\mathbf{v})/\partial b_j = \langle h_j \rangle_{data} - \langle h_j \rangle_{model} \qquad (13)$$

$$\partial \log p(\mathbf{v})/\partial c_I = \langle v_i \rangle_{data} - \langle v_i \rangle_{model} \qquad (14)$$

$$\partial Rs(\mathbf{W})/\partial w_{ij} = \gamma \, (w_{ij} / \surd \Sigma w_{iji}) + (1-\gamma) \, (w_{ij} / \surd \Sigma w_{ijj}) \qquad (15)$$

where, the expectation of distribution p is indicated as $\langle \cdot \rangle_p$. The weights in short rows and columns are reduced to values close to zero after a few hundred iterations, which leads to sparse weight parameters.

The fast adaptive neural network classifier consists of multiple hidden layers that can be trained layer-by-layer. The training algorithm is deep learned by repeating the process several times such that the corresponding parameter $\boldsymbol{\theta}$ is frozen to infer the hidden unit values. It (the inferred values) serves as the input data to train the next higher layer so that the next hidden layer is modeled. Usually, when thresholding and binarization is applied to the derived weight parameters, it results in major drop in classification accuracy from over 90% to less than 50% due to challenges such as, i) exhausting gigabytes of memory (by the weight parameters) along with the bandwidth between the processing units and the memory, ii) causing bottleneck during computation (caused when multiplications is performed between the weight parameters and the input features) iii) implementing high-precision multiplier usually consists of hundreds of logic gates.

On the other hand, the classification accuracy using fast adaptive neural-network classifier reaches around 95%, as the binary weights are achieved by i) deriving the real-value sparse parameters ii) omitting the small absolute values and thresholding the weight parameters iii) binarizing the left out +ve and –ve weights as +1 and -1 respectively. Overall, learning the network architecture using FANNC features two important properties:

i) it implements sparse neural connections which reduce the weight parameters associated with negligible neural connections to zero, which can be omitted.

ii) by representing neural connections as binary integers, as the sparse weights are well separated and hence can be robustly thresholded where, each neural connection can be represented using a single-bit (which saves major proportion of memory relieving the bottleneck).

The DAC_FANNC mechanism predominantly makes use of the classified weighted parameter to predicts the node's resource availability and admit data traffic to reliable node during communication.

3. SIMULATION RESULTS

The Performance of the proposed DAC_FANNC model is analyzed using a prototype developed in MATLAB. Our simulation model takes into consideration set of mobile ad-hoc nodes deployed in area of 100X100 km^2. With radius varying within 100m, virtual clusters are formed. Around 5% - 10% of MANET nodes are considered in each clusters such that it communicates with each other to elect the IACA node (virtual head) and all other nodes in the cluster becomes the registered member of the cluster. Every registered member of the group is directly communicable to its IACA node. Nodes mobility is varied between 0m/s to 25 m/s. During simulation, source nodes are chosen which is set to perform data communication to the destination. The type of traffic considered is classified into two classes: i) the high-priority traffic belonging to class1, ii) the low-priority or best-effort traffic belonging to class 2. Using random CBR connection, the traffic was generated with packet generation rate of each flow or session of 10 packets per second with payload size of 256 byte. Reference Point Group Mobility Model (RPGM) is considered with the desired delivery rate set to be 99% (very high) and 85% (medium). The network model of the DAC_FANNC is shown in figure 8.

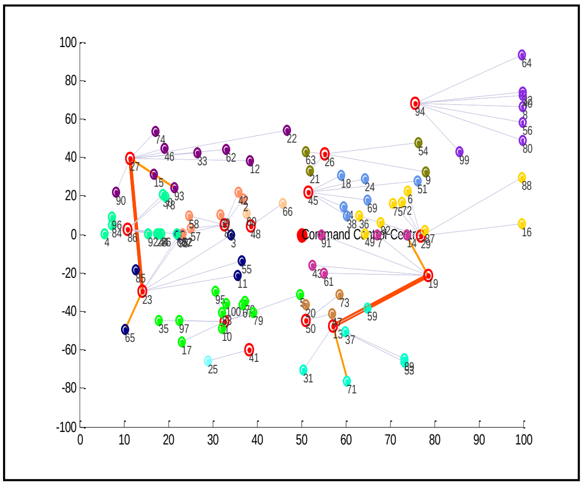

Fig.8. Network model of DAC_FANNC model

To better investigate the scope and performance of our proposed DAC_FANNC approach, it is compared with other similar approaches in terms of throughput and end-to-end delay by, i) increasing the node density or the number of nodes, ii) increasing the node speed, iii) increasing the traffic load or number of sessions in network.

Increasing Node Density:

In this set of simulation, the terrain area, the speed of the node and packets generated by each session is kept constant at 100km^2 X 100km^2, 10m/s and 10 per second, while the number of nodes within the terrain area is alone varied from 100 to 600. The average throughput for high priority and low priority traffic against increasing number of nodes is evaluated and depicted in figure 9 and figure 10.

The average throughput for high-priority and low-priority traffic depicted in figure 9 and figure 10 shows DAC_FANNC scheme outperforms other approaches in successfully delivering packets to the destination. The scheme that is capable of performing an admission control strategy which leads to high throughput specifically for high priority traffic and at the same time satisfying the end-to-end delay requirement incurring low routing overhead is highly preferred.

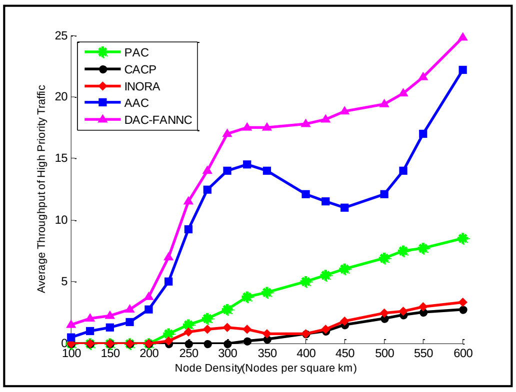

Fig.9. Node Density vs Throughput of high priority packets

It can be observed from the figure 9, that the DAC_FANNC scheme works better specifically for high priority traffic due to the following reason: i) the admission control is enabled and processed only by the virtual head node (the IACA node), rather than by all MANET nodes that reside in the network. ii) the virtual cluster mechanism regulates traffic via the IACA node (among the set of registered nodes) in each cluster rather than every node participating in data communication. iii) registered members forwards high priority traffic to its head which using FANNC mechanism classifies and prioritizes and admits high priority traffic through back-bone links to the destination. iv) the control overhead is low as most control message transmission is handled locally within the clusters rather than across the network. On the other hand, as the network size increases, the relative performance of other schemes such as INORA, DACP, AAC improves to certain extent and then tend to degrade in performance beyond certain limit as it suffers from increased control packets and link breakage due to dynamic changes in topology. Similarly, PAC maintains back up paths which is consistently tested to ensure if it has the end-to-end capacity for the accepted session, which incurs lot of routing overheads causing degradation in throughput. The performance of these schemes decreases as the network size increases due to the fact that the link capacity considered was 2Mbps which causes many bottlenecks in the scenario considered.

For low priority packet transmission, the throughput of DAC_FANNC is found to be better than other schemes as shown in figure 10. As DAC_FANNC regulates high priority traffic through IACA nodes and allows member nodes to handle low traffic messages, handling best-effort traffic incurs low overhead across the network compared to other schemes as high profiled messages are diverted via backbone links.

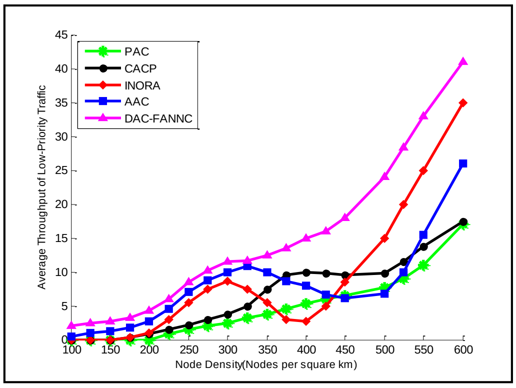

Fig.10. Node Density vs Throughput of low priority packets

As IACA node in DAC_FANNC devotes its resources in performing admission control for high priority traffic, it may not perform equally well for low priority packets. Overall the average throughput of high priority traffic using DAC_FANNC is ~12% - ~15% high compared to AAC approach and greater than ~25% for other approaches while for low priority traffic, the average throughput is found to be ~5% - ~8% better than AAC and ~10% - ~12% better compared to other schemes.

The end-to-end delay for high priority traffic under the scenario where the node density was increased and the behavior of various approaches were observed and the results are referred in figure 11. From the observation in figure 11, the time consumed for transmitting the packet from source to destination is found to be excellent in case of DAC_FANNC model. The end-to-end delay achieved for high priority traffic is sufficiently smaller even in a highly densed network scenario due to less contention resulting due to the reason explained in figure 9. As the network becomes denser, the end-to-end delay is normally expected to increase due to increased contention however, the primary reason for the delay to remain stable and nearly-bounded with small degree of bearable fluctuations is due to the mechanism of handling local task within clusters with the help of IACA node in the network.

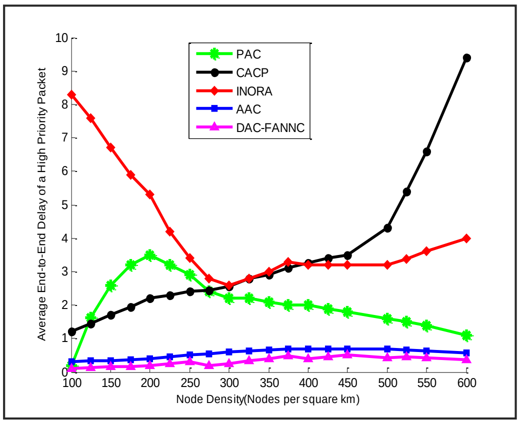

Fig.11. Node Density vs End-to-End Delay

Increasing Node Speed:

In this analysis, the mobility of nodes is increased and the performance of the proposed and existing approaches is assessed. In this scenario, the area of the network is kept constant at 1000 X 1000 m^2 and the number of nodes are kept constant at 300respectively, while the speed is allowed to vary from 0 to 20 m/s, while the pause-time is exponentially distributed with mean value of 30 seconds

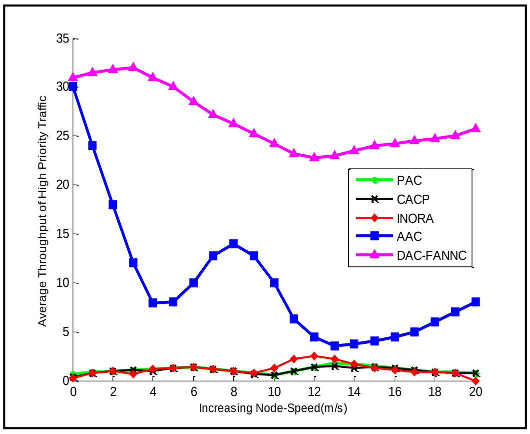

Fig.12. Node Speed vs Throughput of high priority packets

From the figure 12, although the throughput performance is impaired due to increasing mobility, the extent to which it affects the proposed DAC_FANNC approach is low compared to other. The high profiled traffic request from VC_{Node} are sent to IACA which are further classified and admitted as per the network resource availability for efficient management of available resources using FANNC in the proposed approach tends to increase the throughput preventing failures caused due to congestion. Additionally, the cluster formation and managing task locally within clusters ensures low overhead. Significantly, as the virtual head ie, the IACA node takes responsibility to manage high profiled traffic of member nodes, the route discovered by IACA node are sustained for longer duration compared to route discovered between the nodes.

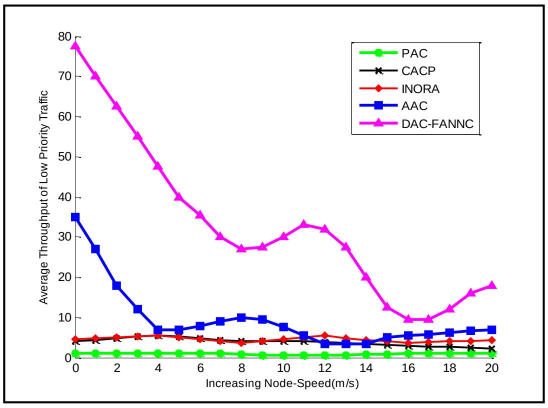

Fig.13. Node Speed vs Throughput of low priority packets

While in other approaches, the maintaining end-to-end routes is expensive as the nodes position keeps changing when speed increases. However, the throughput for low priority traffic as shown in figure 13, is slightly impaired in DAC_FANNC as mobility increases which may be due to errors caused in discovering the next hop member due to high mobility. On an average it was noticed that the performance of DAC_FANNC is ~20% - ~25% more compared to its counterpart for high profiled traffic and around ~10% - ~15% more than its counterpart for low priority traffic when the node speed was increased.

Increase in node mobility tends to dynamically change the topology, which in-turn changes the end-to-end routes discovered by nodes as shown in figure 14.

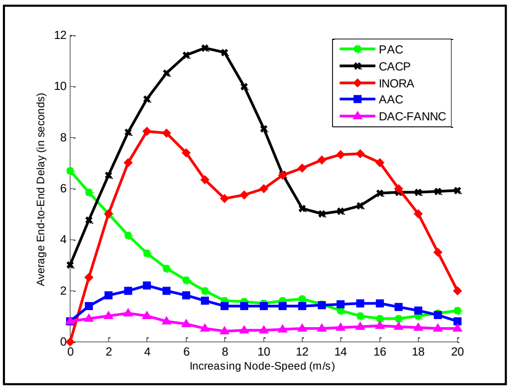

Fig.14. Node Speed vs End-to-End Delay

In case of PAC, if the source node detects change in route discovered or congested paths, then it waits for a back-off time slot and then triggers transmission which leads to additional delay during data communication. This is prevented in case of the proposed DAC_FANNC approach, as the traffic is regulated between the IACA node and member nodes avoiding overheads thus resulting in low delay during transmission even when the mobility of nodes is increased in the network.

Increasing Number of Sessions or Network Load:

By increasing the network-load, the performance of various approaches is analyzed in this scenario. This scenario is inherently used to assess the scalability of DAC_FANNC along with other existing approaches by increasing the number of sessions in the network. By keeping the simulation area, speed of nodes and number of nodes constant such as 600 X 600 m^2, 150 and 10m/s respectively, while the number of sessions is allowed to vary from 8 to 128.

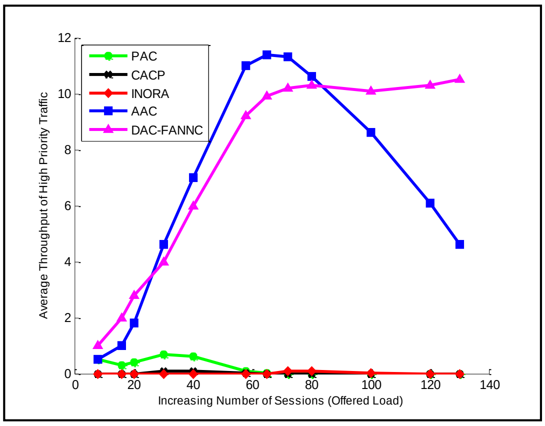

Fig.15. Number of Sessions vs Throughput of high priority packets

The figure 15 displays the average throughput of high priority traffic observed for all schemes under varying load conditions. Due to the desirable features offered by DAC_FANNC, its throughput is found to be higher compared to other schemes. However, as the network-load increases, the relative performance of DAC_FANNC drops due to the following reason, when network load increases, almost every node starts sending traffic which in-turn increases the number of high priority sessions at IACA node. Under the increased network-load conditions, DAC_FANNC controls, prioritizes and admits packets as per the user priority and finds it difficult to satisfy the end-to-end deadline requirement of high-priority traffic and hence is urged to drop packets. In other words, it tries to achieve the bounded delay requirement of high-priority traffic at the expense of throughput degradation. This is partly attributed to the admission control built together with DAC_FANNC, and this plays a significant role in throttling traffic, so that demand can fit into the available capacity.

Figure 16 shows the average end-to-end delay performance of the high-priority traffic when the number of session is increased.

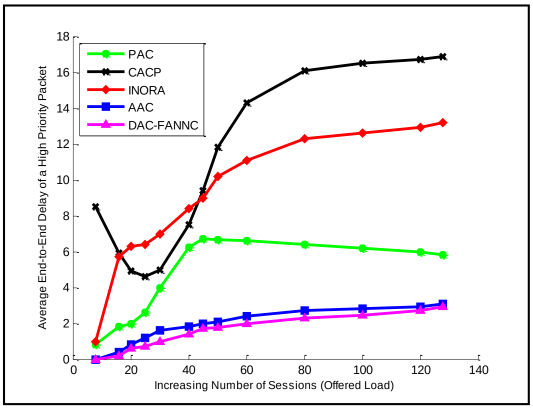

Fig.16. Number of Sessions vs End-to-End Delay

It is observed that when the offered load is increased, existing approaches such as INORA, CACP finds it difficult to keep the end-to-end delay within the bounds due to long delays incurred at the MAC level due to increased contention at the neighbor nodes and maintain the end-to-end paths. With the link capacity of the considered network is 2 Mbps, provided if the packet generation rate of each flow is considered to be 10 packets per second, it in-turn incurs a payload of 256 bytes - obviously, the link capacity poses the main bottleneck, when the offered load increases beyond the network capacity, the intermediate nodes attempts to discard the packets due to long delays in case of existing approaches, which is reduced in the proposed DAC_FANNC with the help of localized service management making it more reliable compared to its counterparts.

4. CONCLUSION

In line with lot of research works involved in MANET AC strategies, in this paper we have proposed a "Distributed Admission Control using Fast Adaptive Neural Network Classifier (DAC-FANNC)" for managing network resources using deep neural network strategy. The proposed model utilizes the advantage of virtual clusters and electing Intelligent Admission Control Agent (IACA) node for localizing task management among cluster members. The distributed IACA node is used to predict dynamic changes and availability of resources and perform admission decision by accurately classifying the node's state using FANNC using the resources such as band width, link expiration time, etc. The IACA regulates and admits high priority traffic through backbone links. Experimental analysis ensures that proposed approach is better compared to its counterparts in dynamically regulating real-time high profiled data traffic and accurately allocating resources in face of network dynamics such as varying node density, load and mobility. In line with the ultimate aim of current research on Admission Control, the objective of future admission control strategies in MANET is to implement AC schemes that are intelligent and can be used for variety of delay-sensitive, interference-sensitive applications and adapting itself to different networks sizes and load factors. The idea for future AC design should have to carefully consider the practical significance involved in efficient use of network resources.

REFERENCES

[1] Zhang, R., & Rubin, I. (2006). Robust flow admission control and routing for mobile ad hoc networks, in Proceedings of the IEEE Military Communications, pp. 1–7.

[2] Khan, A., & Khattak, K. (2010). AC and QAR for provisioning of QoS in MANETs. Master thesis.

[3] Hanzo, L., & Tafazolli, R. (2007). Throughput assurances through admission control protocol for multi-hop MANETs, in Proceedings of the IEEE International symposium. Personal, Indoor and Mobile Radio Communication, pp. 1–5

[4] Akbarzadeh, S., Cottatellucci, L., & Bonnet, C. (2009). Low complexity cross-layer design for dense interference networks, in Proceedings of the International Symposium on Modeling and Optimization in Mobile, Ad-Hoc and Wireless Networks (WiOpt).

[5] Yu, J., & Yand Chong, P. H. J. (2005). A survey of clustering schemes for mobile ad hoc networks. IEEE Communication Surveys Tuts., Vol.7,pp.32-48.

[6] Niu, H. L., & Liu, S. (2017). Novel PEECR-based Clustering Routing Approach. Soft Computing 2017, 21(24): 7313-7323 DOI: 10.1007/s00500-016-2270-3.

[7] Oh, S. Y., Marfia, G., & Gerla, M. (2011). MANET QoS support without reservations. Journal of Security and Communication Networks 4(3), 316–328.

[8] Lajos, H., & Rahim, T. (2009). Admission control schemes for 802.11-based multi-hop mobile ad hoc networks: a survey. IEEE Communications Surveys & Tutorials 11(4), 78–108.

[9] Johnson, D., Maltz, D., & Hu, Y. (2007). The Dynamic Source Routing Protocol (DSR) for Mobile Ad-hoc Networks. IETF MANET Working Group Experimental RFC 4728.

[10] Xiang, X., Wang, X., & Yang, Y. (2010). Stateless multicasting in mobile ad hoc networks. IEEE Trans. Comput. 59(8), 1076–1090.

[11] Abdrabou, A., & Zhuang, W. (2008). Stochastic delay guarantees and statistical call admission control for IEEE 802.11 single hop ad hoc networks. IEEE Trans. Wirel. Commun. 7(10), 3972–3981.

[12] Chakeres, I. D., & Belding-Royer, E. M. (2004). PAC: perceptive admission control for mobile wireless networks, in Proceedings of the IEEE International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks, pp. 18–26.

[13] Yang, Y., Kravets, R. (2005). Contention-aware admission control for ad hoc networks. IEEE Trans. Mob. Comput. 4, 363–377.

[14] Dharmaraju, D., Roy-Chowdhury, A ., Hovareshti, P., & Baras, J. (2002). INORA - a unified signaling and

routing mechanism for QoS support in mobile ad hoc networks, in Proceedings of the IEEE International Conference on Parallel Processing Workshop, pp. 86–93.

[15] Park, V., & Corson, S. (1997). Temporally Ordered Routing Algorithm v1 Functional Specification IETF Internet Draft.

[16] Lee, S., Ahn, G., Zhang, X., & Campbell, A. (2000). INSIGNIA: an IP-based quality of service framework for mobile ad hoc networks. Journal of Parallel and Distributed Computing 60(4), 374–406.

[17] Cano, C., Bellalta, B., & Oliver, M. (2007). Adaptive admission control mechanism for IEEE 802.11E WLANS, in Proceedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communication.

[18] Sanni, K. (2015). FPGA implementation of a Deep Belief Network architecture for character recognition using stochastic computation, Information Sciences and Systems IEEE, 1 - 5.