# Enhancement of SIP over MANET: a hierarchical clustering approach

SALMA R. ABDELHAMID
Faculty of Computers and Information Technology
Future University in Egypt
Cairo, EGYPT
salma.radwan@gmail.com

HOSSAM-E M. SHAMARDAN
Faculty of Computers and Information
Helwan University
Cairo, EGYPT
hossam@fci.helwan.edu.eg

ATEF Z. GHALWASH
Faculty of Computers and Information
Helwan University
Cairo, EGYPT
atef.ghalwash@fci.helwan.edu.eg

*Abstract:* - The field of wireless communications has witnessed an unprecedented growth during the past decades. The presence of such a fast rate development in wireless networking and Internet Technology (IT) devices persuades the researchers to focus on a worldwide used type of networks, namely the Mobile Ad-hoc Network (MANET). A MANET is a special type of decentralized wireless networks consisting of a group of randomly distributed devices with wireless capabilities. This infrastructure-less unique type of networks directed the researchers towards proposing new approaches that facilitate the implementation of the widely used services, protocols, and applications of the wired networks. Among which, the Session Initiation Protocol (SIP) is a signaling protocol used for supporting Voice over Internet Protocol (VoIP) applications. It allows the establishment of multimedia sessions and calls between different parties. SIP functionality totally depends on a centralized infrastructure, and complexity arises when deploying such a protocol over MANETs. This paper proposes a new hierarchical clustering theme for MANET routing. The proposed approach compensates the lack of a centralized infrastructure that allows the deployment of SIP over MANETs. Enhancing routing modules are presented to build the routing tree based on a hierarchical addressing theme.

*Key-Words:* - MANET, SIP, VoIP, Clustering, Routing, Hierarchy

## 1 Introduction

Characterized by its low cost and flexibility, the Mobile Ad-hoc Network (MANET) has experienced an incremental expansion during the past decades. A MANET is a special type of wireless networks that consists of a group of wireless mobile nodes (MNs) such as laptops, or mobile phones connected together through wireless links [1]. Unlike traditional infra-structured wireless networks which mainly depend on centralized entities, MANETs are formed spontaneously and they lack the existence of a centralized infrastructure. In addition to that, they are characterized by a dynamic topology as the network faces rapid and frequent changes due to the mobility of the nodes, or sometimes their failures.

The MNs of the MANET are self-organized and self-configured devices, and are able to communicate together despite this absence of an underlying structure or a centralized administrative support [1, 2].

MANETs play an important role in the wireless generation, and the attempt to deploy internet based applications is dramatically increasing with time. MANETs have a numerous range of application such as military tactical applications that demand highly secured and reliable networks [3, 4]. Other applications might be more sensitive to time delay or bandwidth consumption such as Voice over Internet Protocol (VoIP) applications. The VoIP can be used to deliver voice and video over the internet. VoIP is highly adopted nowadays, and thanks to the presented free services such as video-conferencing, determining and blocking caller ID, or tariff-free long-distanced international calls, VoIP is favored over the traditional Public Switched Telephone Network (PSTN) [5-7].

However, the use of VoIP requires the establishment of a session between end users. And for such establishment, the Session Initiation Protocol (SIP) is considered as a key element.

SIP is an application-layer signaling protocol that is used to control multimedia communication sessions such as initiating, modifying, or terminating an interactive session. In other words, it is a protocol that enables two parties to call each other and to negotiate the parameters of the multimedia session. SIP is totally built upon a centralized infrastructure including different entities such as Proxies and Registrars, typically owned by the network operator. Due to the major contradiction between the decentralized architecture of MANETs and the SIP, the later cannot be directly deployed into MANETs. And since it is an undeniable fact that SIP multimedia services over MANETs can be indisputably exploited in different

business areas, the exploration of its deployment in MANETs is highlighted, and different approaches were proposed [8]. This paper presents a new hierarchical clustering algorithm to enhance the performance of SIP over MANETs.

The rest of the paper is organized as follows. An overview on MANETs is presented in section 2, highlighting some of its routing protocols concepts. SIP, its entities, and functionalities are elaborated in Section 3. Section 4 wraps up the problem of deploying SIP over MANETs. Some of the related work is explained in Section 5. The proposed Algorithm is discussed is section 6. Finally, the overall conclusion is summarized in Section 7.

# 2 MANET overview

The MANET is a collection of self-configured mobile devices with wireless communications and networking capabilities. Instead of relying on centralized entities such as routers and switches, each of these mobile nodes can act as a router, transmitter, or a receiver turning the network to a decentralized wireless network that uses multi-hops wireless links for communication [2].

## 2.1 MANETs characteristics

MANETs have a very unique set of features that contribute in the wide spread of this type of networks. The most explicit feature is that this network works in a distributive manner without the need of centralized entities such as access points [9]. Also, the network consists of low-priced IT devices and eliminates the need of high cost fixed entities. Moreover, since all nodes act as routers at some point, nodes that are out of transmission range can be reached by multi-hops messages that are forwarded through intermediate nodes existing between the source and the destination.

However, the same characteristics outline many obstacles and challenges when using applications or protocols deployed in the wired or centralized wireless networks. Contradicting with traditional wireless networks, the nodes of a MANET do not have fixed locations; instead, they move from one position to another or even leave the network causing an unpredicted dynamic change in the network topology. MANETs also suffer limited bandwidth, asymmetric wireless links, and limited resources that are highly affected by multi-hops transmission. Furthermore, the fading of the signal, noise effect and signal interference degrade the wireless link capacities [1, 2, 9].

## 2.2 MANET routing protocols

The challenging MANET characteristics obstructed the direct deployment of wired and wireless networks protocols. The end users of ad-hoc networks compensate the decentralized infrastructure-less architecture by relying on each other in routing. Each node contributes in routes discovery, and data sending from the source to the desired destination usually require multiple hops.

For a routing protocol to be functional in such an environment, it has to cope with the dynamic topology of the network, as well as the absence of a fixed network infrastructure [10, 11]. And for the protocol to be effective, lots of operations and properties should be supported, amongst which are reactive and demand based operations, distributed operations, multiple routes providence, security measures and looping avoidance [2, 12]. A lot of classifications exist for Ad-hoc networks Routing Protocols [2]. The most common classification is based upon how routing information is acquired and maintained by the nodes of the network. Accordingly, the Ad-hoc routing protocols can be divided into three types: Proactive, Reactive, and Hybrid routing protocols [10, 12-15]

### 2.2.1 Proactive (table-driven) routing protocols

In proactive routing protocols, the nodes keep up-to-date routing information so that the packet is directly forwarded when required to the destination. Routing tables are used to maintain the routes to all possible reachable destinations. These tables are periodically updated and sequence numbers are used to disseminate fresh routes in case any change occurs in the topology of the network. The routing protocols based on this methodology differ in the way nodes update the topology change of the network and the routing information maintained in the routing tables.

However, this type of protocols does not perform effectively in highly dynamic networks as for each topology change; the MNs are obliged to update their routing tables causing an increase in the control message overheads thus leading to an overall network performance degradation [13]. Examples of this protocol are Destination sequence Distance Vector (DSDV) [16-18], Wireless Routing Protocol (WRP) [16, 19] and Clusterhead Gateway Switch Routing [2, 20].

#### 2.2.2 Reactive (On-Demand) routing protocols

In reactive routing protocols, the routes to the destinations are not gathered a priori. The route discovery process takes place only when a node "demands" the route to a certain destination. Upon its desire to send data to a certain destination that it doesn't know its location, the source node applies the route discovery mechanism which floods the network with route requests messages (RREQ) until the destination is reached or a node that has a fresh route to the destination replies back with a route reply (RREP) message [13]. Reactive protocols overcome the large memory consumption needed in maintaining routing tables of proactive protocols, and no periodic tables update is applied. However, the drawback of reactive protocols is the delay caused during the route discovery process which makes it inconvenient for time sensitive applications. An example of this protocol is the Ad hoc On-demand Distance Vector Routing (AODV) [12, 21, 22], and Dynamic Source Routing [15].

#### 2.2.3 Hybrid routing protocols

As concluded from its name, a hybrid protocol is a merge between both proactive and reactive protocols. The routing is initially established using the proactive approach, and then the route requests from additionally added nodes are provided using the reactive approach. This approach tends to reduce the overhead of the proactive protocols as well as the time delay of the reactive protocols [2, 13]. An example of such protocol is the Zone Routing Protocol (ZRP) [23].

## 3 SIP overview

Standardized by the Internet Engineering Task Force (IETF), SIP is a text-based and HTTP-like application-layer signaling protocol that is used for controlling multimedia communication sessions between end users [24]. Independent on the underlying transport layer, SIP is a client-server based protocol that is used to establish and control multimedia sessions between users. SIP is responsible of determining user location, confirming, or denying its availability and due to its flexibility and simplicity, SIP has become the most widely used protocol in VOIP networks [26].

### 3.1 SIP architecture

In its functionality, the SIP depends on a centralized architecture. The SIP entities are classified into two types, user agents and SIP servers [26]. The user agents are the end points of the network such as mobile phones, Personal Digital Assistant (PDA), or laptops. There are two types of user agents, the User Agent Client (UAC), which is the end point that initiates the calls and session establishment requests. The other type is the User Agent Server (UAS). It is the agent that receives the SIP requests and replies back with SIP responses. Examples of SIP methods are REGISTER, INVITE, UPDATE, CANCEL, ACK, or BYE [27].

SIP servers on the other hand are divided into four main types; Proxy Servers, Registrars, Location Servers, and Redirect servers [27, 28]. To be able to make a call, the UA must first register itself at the registrar. The registrars then store the locations of the registered users at the location servers. The proxy server is an entity responsible of making requests on behalf of the user agents. It can act both as a client or a server. It is also responsible for assuring the authentication of the end user and whether it is allowed to make a call or not [29]. The redirect servers are responsible of redirecting SIP users to another entity allowing them to connect to different set of addresses to reach the required destination.

### 3.2 SIP functionality

As previously mentioned, the main role of the SIP is to establish a multimedia session between the end users; afterwards the actual data flow is directly carried out between those users using the underlying transport layer. To identify SIP users, each of them is given a SIP address-of-record (AOR) Uniform Resource Identifier (URI). This address can be resolved at the SIP proxy of this users' domain. Each user is supposed to register its address, using the SIP method REGISTER, at the SIP domain registrar, and thus allowing the identification of this user's actual location in terms of IP addresses [30].

The nodes of the networks exchange different SIP messages, either requests or responses. When a node wants to initiate a connection, it sends an initiation request INVITE to its local proxy server without knowing the exact location of the called party. The proxy will determine the route to be taken to the callee after consulting its registrar and location server. The INVITE request is then forwarded by the proxy, and possibly through other number of proxies, until it gets to its destination. The callee could either accept or reject the incoming call. If accepted, the session initiation is then finalized by having the caller acknowledging the reception of the callees answer [27, 30]. After finalizing the session establishment, and as

mentioned earlier, the data exchange between the two parties takes place without the involving of SIP proxy, and using the agreed upon transport protocol.

Fig. 1 illustrates a simple voice connection using SIP. The two involved end users have to first register with their usernames at each one's proxy. After receiving an INVITE request from the calling UA, the SIP proxy looks up the URI of the destination, and then forwards the INVITE message to the UAS based upon the acquired addresses, or set of addresses in some cases [28, 31] .

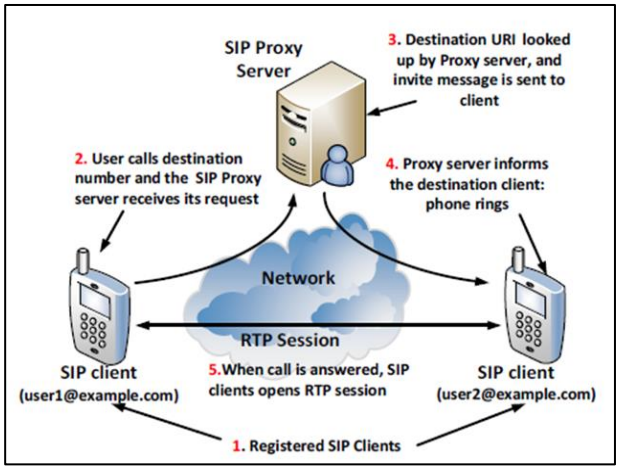


Fig. 1, Basic SIP call flow [5]

# 4 Problem definition

Ad-hoc networks have a great advantage of designing flexible networks. However, as mentioned earlier, the mobility of the nodes and the lack of a centralized infrastructure remain a challenging constraint in some research areas. Due to such constraints, the nodes are not familiar with the topology of their networks. Instead, they have to enter a discovery phase first. When joining the network, each node should be able to announce itself and discover neighboring nodes to be able to send its data to the required destination. Afterwards, routes should be discovered and transmission takes place using cooperative multi-hops between source and destination. Moreover, this dynamic un-centralized architecture makes it difficult to implement any form of communications that requires session establishments instead of the best-effort dissemination of independent packets from and to multiple nodes.

Contradicting with the dynamic and decentralized features of the MANET, SIP relies in its functionality on centralized entities. Each entity has a specific role in setting up the connection and defining the routes between sources and destinations. This centralized SIP architecture is obviously not straightforwardly applicable to the MANET. SIP users in MANETs cannot reach other parties, as they do not have support from proxy servers. Moreover, they cannot be reached by other nodes, as there are no SIP registrars where they can register their contact information. Nevertheless, the Quality of Service (QoS) metrics of VoIP such as bandwidth consumption, delays, jitters, packets loss, and signaling are severely degraded as the traffic and number of multiple hops to the gateway increase. Consequently, several problems arise when directly deploying SIP services in ad-hoc networks, and altering the SIP main centralized architecture should first take place.

# 5 Related work

## 5.1 Cluster-based routing protocols

Cluster-Based Routing Protocol (CBRP) is a cluster on-demand source routing protocol [32, 33]. In this routing protocol, the nodes of the network are aggregated into overlapping or disjoint clusters. Each cluster comprises a group of nodes; each of which is assigned a different role or status. The node can be a Cluster Head (CH), Cluster Gateway, or an ordinary Cluster Member. The structure and members of a clustered network is shown in Fig. 2.

The Cluster head (CH) is the coordinator of the cluster. It is the node responsible of managing the cluster and the inter-cluster communication. The CH is also fully aware of its group members and link state information in the cluster [32]. This CH is elected based on several proposed techniques [34-36]. Nodes having the status as Gateways are those members which are in the range of two or more different CH. They are contacted by CH for inter-clusters communications. The rest of the nodes are considered ordinary nodes or members that build their communications through their corresponding cluster heads.

The nodes of the network maintain the information about their neighbors in routing tables, along with the status of each neighbor, and Hello messages are used to periodically update the existence of nodes and links. Moreover, this type of routing makes use of sequence numbers to distinguish between data of old routes and fresh ones. Cluster based routing protocols depend on routing between clusters instead of nodes, thus the overall overload, scalability and throughput is enhanced [2, 34].

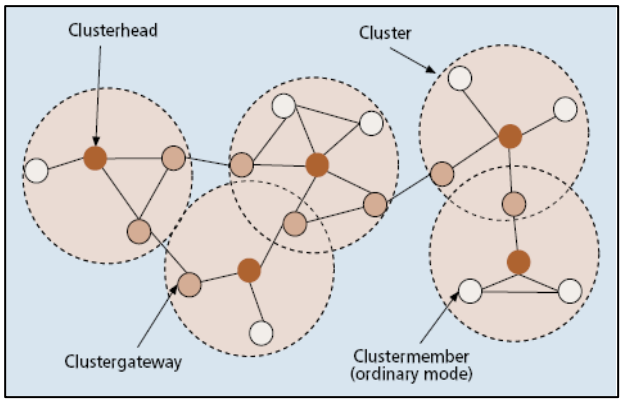


Fig. 2, Clustering structure [37]

To discover a route, if a source (S) wants to send data to a destination (D), it first sends a Route Request (RREQ) packet to its CH recording only itself in the source route. The CH checks its table to determine whether this destination is a member of its own cluster or not. If (D) is in its cluster, then the CH forwards the RREQ to the destination by appending itself in the route. But if the destination is not in the same cluster as the source, the the CH forwards the packet to the the neighboring cluster heads or gateways. Any node that forwards this packet also adds its ID in this RREQ. To provide loop-free routes, the packet is only forwarded by the node once, and is not forwarded to a node with ID appearing in the recorded route [32, 33]. Finally, when the request reaches the required destination, node D sends back a Route Reply (RREP) packet including the complete reversed route of that received in the RREQ. Of course intermediate cluster heads and gateways forwarding the RREP will memorize this route for future route discoveries. Route Error (RERR) packets are also used by this protocol to report an error in the routes or wireless links between nodes.

## 5.2 Deployment of SIP over MANET

This sub-section presents some of the proposed approaches addressing SIP signaling over MANET.

Leggio et al. proposed in [38] a decentralized approach. This approached aimed to allow the deployment of SIP in MANETs by embedding a limited set of SIP server functionalities into SIP user agents. By such approach, the operations originally carried out by centralized SIP entities is merged in all end users in the ad-hoc networks, and each node carries out the required function in order to establish the session.

To discover the users AOR, the authors present two methods, Fully Distributed SIP (dSIP) and SIP with Service Location Framework (sSIP). Needless of centralized SIP units, dSIP depends on SIP methods to acquire user contact information. When a broadcasted REGISTER message is received, it is stored in the local cache of the node, and when an INVITE message is received, the local proxy modules plays the role of binding between the specified address and SIP AOR. sSIP on the other hand depends on Service Location Protocol (SLP) [39]. In this method, each node receiving the broadcasted SLP request replies back with SLP reply including its binding. This approach suffered a scalability problem as the message broadcasting leads to high bandwidth consumption and flooding in large MANETs.

SIPHoc, a middleware infrastructure that is used to handle and manage the session setup was proposed by Stuedi et al. [31]. SIPHoc is a completely decentralized approach, which eliminates the need of centralized components, or assigning certain functionalities to some elected nodes of the network. SIPHoc is independent on the underlying network topology, supports static and mobile ad-hoc networks, and even allows the use of SIP applications without any modifications. Furthermore, this approach is based upon MANET SLP for registration and look-up operations which relies piggybacking techniques that provide an efficient messaging system. The simulation of using SIP applications with this approach resulted in optimum and comparable overhead with the standard MANET operations.

H. Chu and W. Chen proposed in [40] an integrated middleware to take over the roles and functions of SIP servers. The user agents register themselves at this middleware instead of the registrar server. The authors used AODV as an underlying routing protocol, and the exchanged SIP messages are delivered by this protocol. The authors refined the registration process by forcing the middleware to act as a registrar server, thus locally binding the IP address with the AOR. Moreover, the same middleware is responsible of forwarding the SIP messages on behalf of the SIP proxy server. The goal of this approach was enabling SIP applications in MANETs in addition to minimizing the signaling overhead in the network.

In [41], Mourtaji et al. proposed building a Virtual Network for Session Initiation Protocol (VNSIP). The main idea was to self-organize the

Ad-hoc Network using a virtual backbone. The virtual network was constructed using specific nodes within this network. The functionalities of the SIP servers are to be embedded into all nodes of the MANET, and according to a Virtual Network Algorithm (VNA); each node gets the function of a certain SIP server activated so that at the end, all SIP servers would be included in the network to accomplish the address distribution task. During the registration process and depending on whether the node is a member of the VN or not, the SIP are either locally sent to the node itself or broadcasted to all of the 1-hop neighboring nodes. This broadcasting is known as the Replication Mechanism. The replication mechanism allows searching in multiple SIP Proxies at the same time, and thus ensuring short response time. The simulations prove that the performance of VNSIP was remarkable at the session establishment time and failure tolerance, but when it comes to Bandwidth consumption; the performance is unsatisfactory. This behavior is a result of the high number of exchanged replicated SIP messages.

An enhancement was later proposed to overcome this unsatisfactory bandwidth consumption [42]. The proposed algorithm, MANET Call Admission Control (MCAC), is used to permit the establishment of calls to a number that agrees with the available bandwidth of the MANET, otherwise it rejects the calls based on a certain threshold. When this threshold is exceeded, the establishment of new sessions is refused, and thus the already established communications remain undisrupted.

Almobaideen et al. presented a Fuzzy and Cluster based SIP Protocol (FCSIP) [25]. FCSIP is an application layer protocol that is independent on the underlying routing protocol. The MANET in which this protocol is applied is assumed to be clustered. Aside from several used algorithms for electing the CH [34, 35], the authors rely on the VoIP activeness of the node to be elected as a cluster head. The main advantage of using an underlying cluster-based routing protocol is the significant reduction of transmitted SIP control messages, and this comes as a result of the hierarchical level represented by the CH and the members falling beneath it. The enhancement is proven by the simulation that compared FCSIP to a fully distributed version of SIP which they refer to as FDSIP.

# 6 The proposed approach

As previously mentioned, a lot of the MANET routing protocols suffer a scalability problem and a large overhead traffic [16]. Thus, this proposed approach depends on the Cluster-Based Routing Protocol (CBRP) [32, 36]. The clustering-based algorithms have proven to improve the flexibility and scalability of the network as well as the utilization of the bandwidth [2, 34]. in addition to the clustering of the network nodes, this approach also proposes a hierarchical-ID clustering theme that facilitates the routing among the nodes and reduces the overhead traffic. The main idea is that each CH elects another Child CH (CCH) in the neighboring clusters and assigns a specific address to it that extends the address of the parent. The following sections represent a detailed discussion of the algorithm.

## 6.1 Cluster formation

The first step of the cluster formation is selecting a node from the network to be the CH of the root cluster. Many algorithms were proposed for such a selection [34, 35]. Our approach selects the root as the node having the highest connectivity among its neighbors. This node broadcasts a message announcing its desire to form a cluster. The broadcasted message includes a parameter that represents the maximum number of hops for the message to be forwarded, or in other words the message max Time-To-Live (TTL_max). Each time the node is forwarded, its Time-to-Live field is decremented by one till it reaches TTL_max, afterwards the packet will it will die out. Another included parameter is one that defines the cluster coverage; the max number of hops between the members of the cluster and their CH (hops_max). Nearby nodes that are within this hops count will reply to the broadcasted with an acknowledgment (ACK) message and will join the cluster and send their parameters. For our approach, hop_max is set to 1, which means that all members are only one hop way from their CH. Other parameters that will be used in our proposed routing and addressing themes are also exchanged. Such parameters are the identifier of the sending cluster head (CH_ID), and the depth of the cluster (d) in the network.

When acknowledgments within a certain allowed time (ack_timeout) are received, the initiating node sets its Cluster Head Flag (CH_Flag), and updates its nodes list (members_list). Among this list, and based upon the number of hops (hops) and the Received Signal

Strength Indicator (RSSI), several children are elected by the CH to be new descendent cluster heads that again initiate forming another level of clusters. This cycle is carried on until all nodes of the network are within the formed clusters. The proposed clustering algorithm is shown in Fig. 3.

```
Form_Cluster (CH_ID, delay, d)
{
  Wait (delay)
  Set hops_max to 1;
  Set TTL_max to 2;
  Set max_childs to required number ;
  Bcast_Cluster_request (CH_ID, hops_max, TTL_max, TTL, d);
  members_list ←received_ACK (NID, hops, RSSI , ack_timeout);
  if (members_list = = NULL)
      {
        Join_Cluster ( );
      }
   else
      {
       Set CH_Flag to 1;
       for i = 1  to  max_childs
         {
          CCHi = Select_CCH_node  (members_list) ;
          Assign childi its CHi_ID from list of available addresses;
          Set delayi
          Set di of CCHi to d+1;
          Request from CCHi to form_cluster (CCHi_ID, delayi, di);
         }
      }
}
```

Fig. 3, Algorithm of forming the cluster

In addition to the proposed hierarchical clustered tree, when the CH assigns an address to its CCH, the former merges its own address to the new assigned one, in addition to another new part representing a unique address of the selected child. Following this approach, each CH will keep track of its descendant CHs facilitating the route discovery procedure and decreasing the flooding of route requests messages. Further elaboration is presented in the hierarchical addressing theme (section 6.3).

## 6.2 Cluster joining and child electing

Upon receiving a broadcasted cluster formation request, the node first checks whether its cluster head ID (my_CH_ID) is assigned or not. If it is assigned with an ID, this means that the node already belongs to another cluster and will not reply with an acknowledgment message. Still, the parameters of the requesting node will be registered in its routing table. On the other hand, if the node does not belong to any cluster and is within the specified hops_max, it sets the ID of its cluster head, and its own depth (my_d) to the corresponding parameters sent in the broadcasted request. An ACK is then sent to the requesting CH indicating the joining node ID (my_NID), hops count, and the RSSI. At this point, the joining node does not have an assigned Internet Protocol (IP) address, so my_NID is generated using its MAC address [43-45]. The node also checks the TTL_max field sent by the CH, if it is still valid, then the former forwards the broadcasted message to its neighbors.

After the reception of the ACK messages sent by the joining nodes, the CH checks the registered parameters of the node. The CH is mainly interested in two parameters; the number of hops that lie between the CH and the joining node and the RSSI. The number of hops between the CH and the responding node is bounded by the TTL_max field previously mentioned. TTL_max defines the overlapping level of clusters. If for example this parameter is set to be equal to our defined hops_max which is 1, this means that the selected CCH will lie in the same cluster of the broadcasting CH. And based upon the RSSI, the CH will elect a child that lies at the boundary of the cluster. This scenario is known as the Simple Hierarchical clustering (SHC) [46]. Such election will result in overlapping between the clusters. If TTL_max is set to a value greater than hops_max, this means the elected children will lie outside the cluster and overlapping will be reduced. And as the overlapping of the clusters decreases, the breadth and the depth of the tree decreases, which in turn enhances the latency and energy consumption required to deliver a message. Furthermore, non-overlapping clusters provide a better load balancing [46].

In our approach, TTL_max is set to 2 max_hops, which is equal to 2. This means that when nominating a CCH, the CH looks at the nodes which are 2 hops away from it to be its descendent CHs, and nodes that are one hop away will be cluster members. The CH will then send unicasted requests to the selected CCHs, and each of them will wait for a random delay before replying with an ACK and staring forming the subsequent level of clusters. This random delay is to prevent candidates from forming clusters at the same time, thus reducing the chances of two CCH forming nearby overlapping clusters. During the waiting time, each candidate remains listening to any ACK meant to the same CH. If any ACK is heard, the candidate abandons its candidacy to be a CH and goes for joining a cluster. If no ACK is detected during that time then the node sends an ACK to the parent CH and waits for an order from the CH to start forming a new inherited cluster. The algorithm of joining the cluster is illustrated in Fig. 4.

```
Join_cluster( )
{
  Listen to Bcast_Cluster_request (CH_ID, hops_max, TTL_max, TTL, d);
  TTL = TTL– 1;
  hops = TTL_max – TTL;
  if (hops ==1)
 {
    If (my_CH_ID = not assigned)
     {
       Set my_CH_ID to CH_ID;
       Set my_d to d;
       Send ACK (my_NID, hops, RSSI);
     }
    else
     { Add CH to table;    }

    Wait (random_delay);
    Forward Bcast_Cluster_request (CH_ID, hops_max, TTL_max, TTL, d);
    Exit();
 }
  else if (hops==2)
 {
        Wait (random_backoff);
        If ( detect_neighbour_ack = = TRUE )
           {   Join_cluster();   }
      else
         {
            Send ACK (my_NID, hops, RSSI);
            Wait for CH reply;
            If (received form_cluster (CH_ID, delay, d) = = TRUE)
               {   Form Cluster (my_N_ID, delay, d);
                   Exit ( );
               }
         }
 }

}
```

Fig. 4, Joining the cluster algorithm.

In order to optimize the hierarchical tree, each CH and starting from the root node broadcasts its depth to the neighboring nodes. CH nodes hearing this broadcast will check their depth, and if it is greater than the depth broadcasted, they send to the broadcasting node requesting the later to accept them as children. As a result, if the requesting node is accepted as a child, it will re-execute the form cluster function and reorganization to all involved nodes takes place.

### 6.3 Hierarchical addressing theme

Different addressing themes were presented to resolve the addressing assignment in a dynamic topology network like the MANET [43-45]. And in general, the IPv4 addressing scheme is used in ad-hoc networks [47]. Our approach is inspired by the Logical Hierarchical Addressing (LHA) protocol illustrated in [48].The IP address is split into different partitions; each resembling a certain level and inheritance of the node. The number of bits of each partition defines on the number of clusters allowed to be formed within the network, and the maximum number of nodes that can exist under the control of a single CH. For example, if the node partition is set to n-bits, then this means that the maximum number of nodes that can lie in the same cluster is $2^n$. The approach also assumes that all nodes in the network have the authority to assign an IP address to other nodes; however the address assigning process is only carried out by a CH node.

When a node joins a cluster by sending the ACK to the CH, the CH in turn replies with a unicasted message that assigns a unique address to this node. Part of the address is the same as the CH itself and the other part is a unique identifier that cannot be repeated in the cluster. Every time the CH assigns an address to the node, it first checks in its list of available addresses, and among which the address is selected. Applying this approach allows each CH to keep track of its children and node member forming by so a traceable hierarchical tree for all clusters of the network.

Fig. 5 illustrates the hierarchical addressing theme. The IP-address is divided into two partitions; one for the CH_ID and the other for the ID of the member node. In our approach, the most significant 24 bits represent the complete address of the CH, which will be the same for all nodes existing in the same cluster. Whilst the least 8 significant bits represent a unique address identifier for each node, this address cannot be repeated in the whole network. Since the forwarding function is only executed by the CHs, thus the routing mainly depends on the CH_ID partition only, and when this CH is reached the data can be easily forwarded to any destined member existing in this cluster. The corresponding hexadecimal digits of the CH_ID partition are used, and for simplicity; only 3 hierarchical levels are represented.

The selected root CH is assigned the very first IP address which is 0.0.0.0 and members within this node will have the addresses in the range from 0.0.0.1 to 0.0.0.255. When selecting its children, the CH assigns them a range of CH_IDs that is a merge of its own address in the LSB with the new assigned address in the higher significant bits. Applying so, the root node with hexadecimal ID equals $(000000)_{16}$ will have available CH-IDs in the hexadecimal format $(000010)_{16}$ to $(0000F0)_{16}$, and each of the selected children assigns its members with the same CH address in the CH_ID partition, as well as the available addresses in the 8-bits member partition mentioned earlier. Following the same steps as its parent, the CH of ID $(000010)_{16}$ assigns by turn its selected CHs addresses in the range of $(000110)_{16}$ to $(000F10)_{16}$ This hierarchical addressing narrows down the RREQ forwarding process, as each node is able to identify its own parent and descendants. Moreover, from the destination address, the CH is also able to identify the root parents of the required node and upon

which the decision of forwarding a packet or discarding it is taken. Further illustration and routing algorithms are presented in the following routing section.

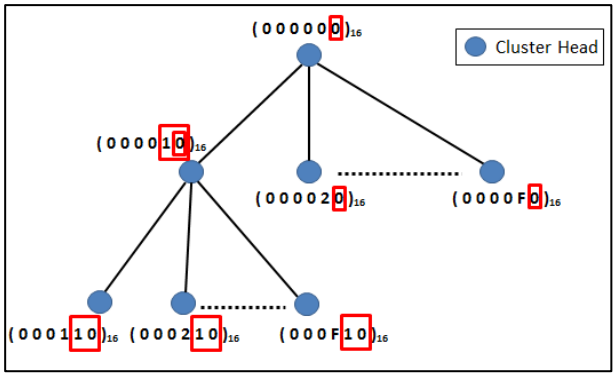


Fig. 5, Hierarchical CH addressing

## 6.4 Routing approach

In conventional cluster-based protocols, when a source node wants to send data to another node, it first has to forward this packet to its CH. If the destination is within the same cluster, or the CH knows a fresh route that leads to this destination then the node forwards the packet to the intended node. But if the CH does not know the route to the destination, it starts broadcasting the route request messages (RREQ) to the neighboring CHs and gateways asking about the path to be taken to reach this destination. When the required destination is reached, route reply messages are sent out and an update in the routing table data takes place [36]. Our proposed hierarchical theme is to enhance the bandwidth consumption by adding some limitations to the forwarding process. Before forwarding the packet to the unknown destination, the CH first checks this enclosed destination address and starts comparing it to its own. The comparison is applied to the 24 most significant bits that define the CH of the node and its location in the tree.

A general adopted algorithm is given in Fig. 6. The decision on forwarding the packet depends on comparing the CH_IDs of the source and destination. If the CH possesses a fresh route to the required destination, then it directly follows the routing process without sending any route requests messages. Otherwise, the CH compares its own CH_ID to that of the destination; if both partitions are alike, then this destination lies in the same cluster, and the CH of this cluster is to directly forward the packet to the destination without the need of any other further comparisons. But if they do not match, or the node doesn't know any fresh routes that lead to this destination, then starting from the least significant digit, the number of matching digits is to be determined. These matching digits represent the address of the common CH between the 2 nodes. If the number of matching digits is greater than the depth of the source CH then the packet is to be forwarded to one of its children. Otherwise, the packet is to be forwarded to the parent CH which re-runs the algorithm again.

For instance, consider a source node of IP address 0.6.160.28 that wants to send data to a destination of IP address 3.214.160.74. The hexadecimal representations of the CH of the source and destination are $(0006A0)_{16}$ and $(03D6A0)_{16}$ respectively. Putting aside our hierarchical addressing, the source would have only known about its parent CH and its direct children, and would have not known the route to the destination thus will forward the packet to its parent CH that will forward it by turn to its parent CH causing unnecessary flooding of RREQ messages. However, applying the hierarchical addressing algorithm allows the CH to have a wider perspective. By comparing the CH_ID of the destination and its own, and determining the number of matching digits, the source discovers that this packet targets one of its descendants. Knowing so, the source does not forward the packet to its parent; instead it forwards it to its child holding the CH_ID $(00D6A0)_{16}$, which in turn forwards it to its child of CH_ID $(03D6A0)_{16}$ thus reaching the cluster in which the destination exists. The CCH then directly forwards the packet to the node of ID 74.

```
check_dest_cluster (current _CH, destination_CH)
{
 if (route to destination is known )
    { execute routing(destination);    }
 else if (current _CH == destination_CH)
    { CH unicasts the packet to its destined member;    }
 else
    {
      n =number of matching LS_hex_digits (current _CH,
destination_CH);
      if ( n < depth (current _CH) )
        {
          Forward packet to parent_CH of current_CH;
          check_dest_cluster (parent_CH, destination_CH);
        }
      else
        {
          next_CH_length = n + 1;
          next_CCH = starting from LSB of destination_CH, get ID
          of length (next_CH_length);
          Forward packet to next_CH;
          check_dest_cluster (next_CH, destination_CH);
        }
 }
```

Fig. 6, Route discovery algorithm

### 6.5 Cluster maintenance

In order to maintain the consistency of cluster information, periodic Hello messages are exchanged between the nodes existing in the cluster and their CH. If any CH detects the absence of one of its member nodes, this node is removed from the members list and its address is added to the list of available addresses so that it is given to any new joining node in the future.

A new node entering the network will attempt to join a cluster by sending a Hello message to announce its existence. The node then waits for any reply from a neighboring cluster head, and then joining the nearest available cluster process and the address assignment continue as previously explained procedure. It is the CH role to decide whether this new node can be a candidate child or not depending on the number of children already existing and the RSSI. Also a node moving from one cluster to another is to update its CH address to that of its new head.

On the other hand, when a CH exits the network, one of its members takes over its cluster head role, using the lowest ID algorithm. The new node also takes the address of the leaving CH and and starts broadcasting its control messages to acquire all necessary data regarding its neighbors and members.

### 6.6 Clustering and SIP functionalities

For deploying the SIP functionalities over the proposed hierarchical clustered network, the widely used decentralized approach is to be adapted to a certain degree [38]. The SIP functionalities are embedded in all nodes of the network; however the server functionalities are only activated when the node is acting as a CH (CH_Flag=1). The member nodes send REGISTER messages to the embedded registrar in their CH to announce their existence as SIP agents, and the CH will play the role of the SIP registrar in binding the IP address of the node to its SIP username. The CHs also takes over the proxy server functionalities in forwarding the UAC messages and the redirect server functionalities in providing to the UAC the IP-address of the required destination. The SIP request messages are carried within the RREQ messages and RREP messages used in discovering the route to a certain destination, thus the IP-addresses of the source and destination URI is determined along with the route. Since the URI is translated to the proposed hierarchical addresses, the path is determined as previously illustrated. The source in such a case locally caches the bindings, and starts its unicasted SIP messages through the determined route. In addition to that, all CH nodes that contribute in forwarding these messages will recognize the bindings as well which will lead to SIP users awareness among the network.

## 7 Conclusion and future work

Being a self-configurable, simple, and flexible wireless network, the MANET has experienced a remarkable evolution over the past years. Researches have been explored to deploy a wide range of applications and protocols over this unique type of networks. This paper proposed a clustering decentralization approach that allows the deployment of the worldwide used protocol for VoIP applications, SIP. The inherently centralized SIP cannot be directly used in the dynamic, decentralized, and infrastructure-less MANET. The approach relied on decentralizing the SIP servers' functionalities into all the nodes of the network based upon their role in the cluster. Moreover, to enhance the routing overhead, a hierarchical CH addressing theme was proposed. This hierarchical addressing set boundaries on the route requests forwarding process and allowed each cluster head to keep track of its descendant CH and nodes. The destined cluster location in the hierarchical tree could be determined by checking the required.

As a future work, we are currently conducting simulations of the proposed algorithm along with comparisons between other approaches to determine the overall performance of the network. And incontestably, the future work in this field of interest is highly promising.

*References:*

[1] D. Ahmed and O. Khalifa, "An Overview of MANETs: Applications, Characteristics, Challenges and Recent Issues", *International Journal of Engineering and Advanced Technology (IJEAT)*, Vol. 6, Issue 4, 2017, pp. 128-133.

[2] D. Ahmed and O. Khalifa, "A Comprehensive Classification of MANETs Routing Protocols", *International Journal of Computer Applications Technology and Research (IJCAT)*, Vol. 6, 2017, pp. 141-158.

[3] N. Raza, M. Aftab, M. Akbar, O. Ashraf, and M. Irfan, "Mobile Ad-Hoc Networks Applications and Its Challenges", *Communications and Network*, Vol. 8, 2016, pp. 131-136.

[4] L. Raja and S. Baboo, "An Overview of MANET: Applications, Attacks and Challenges", *International Journal of*

*Computer Science and Mobile Computing (IJCSMC)*, Vol. 3, Issue 1, 2014, pp. 408-417.

[5] L. Jallow, I. Hwang, A. Nikoukar, A. Liem, "A SIP-based VoIP Application in Enhanced Ethernet Passive Optical Network Architecture", *Proceedings of the International Multi-Conference of Engineers and Computer Scientists (IMECS)*, Vol. 2, 2014.

[6] A. Aburumman , K. R. Choo, and I. Lee, "Nomination-based Session Initiation Protocol Service for Mobile Ad Hoc Networks", *22nd National Conference of the Australian Society for Operations Research (ASOR)*, 2013, pp. 149-155.

[7] S. Kale and V. Khairnar, "SIP Implementation for Ad-hoc Network", *International Journal of Computer Science & Engineering Technology (IJCSET)*, Vol. 6, Issue 6, 2016, pp. 191-194.

[8] A. Hacha, S. Bah and Z. Bakkoury, "A New Cluster-Based Paradigm for SIP Routing in MANET", *Proceedings of the Mediterranean Conference on Information & Communication Technologies (MedCT)*, Vol. 2, 2016, pp. 211-220.

[9] P. Kaur and Sukhman, "An Overview on MANET- Advantages, Characteristics, and Security Attacks", *International Journal of Computer Applications, 4th International Conference on Advancements in Engineering & Technology (ICAET)*, 2016.

[10] S. Lalar and A. Yadav, "Comparative Study of Routing Protocols in MANET", *Oriental Journal of Computer Science & Technology (OJCST)*, Vol. 10, No. 1, 2017, pp.174-179.

[11] A. Gupta, H. Sadawarti and A. Verma, "Performance analysis of AODV, DSR & TORA Routing Protocols", *International Journal of Engineering and Technology (IJET)*, Vol. 2, No.2, 2010, pp. 226-231.

[12] R. Jha, P. Kharga, "A Comparative Performance Analysis of Routing Protocols in MANET using NS3 Simulator", *International journal of Computer Network and Information Security (IJCNIS)*, Vol. 7, No. 4, 2015, pp. 62-68.

[13] A. Hinds, M. Ngulube, S. Zhu, H. Al-Aqrabi, "A Review of Routing Protocols for Mobile Ad-Hoc NETworks (MANET)", *International Journal of Information and Education Technology (IJIET)*, Vol. 3, No. 1, 2013, pp. 1-5.

[14] Dhenakaran and A. Parvathavarthini, "An Overview of Routing Protocols in Mobile Ad-Hoc Network", *International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE)*, Vol. 3, No. 2, 2013, pp. 251-259.

[15] G. Walia, "A Survey on Reactive Routing Protocols of the Mobile Ad hoc Networks", *International Journal of Computer Applications (IJCA)*, Vol. 64, No. 22, 2013, pp. 45-51.

[16] S. Al-Omari and P. Sumari , "An overview of Mobile Ad-hoc networks for the existing Protocols and Applications", *International Journal on applications of Graph Theory in Wireless Ad-hoc Networks and Sensor Networks (GRAPH-HOC)*, Vol. 2, No. 1, 2010, pp. 87-110.

[17] J. Bhatt and N. Hemrajani, "Effective Routing Protocol (DSDV) for Mobile Ad Hoc Network", *International Journal of Soft Computing and Engineering (IJSCE)*, Vol. 3, Issue 5, 2013, pp. 4-7.

[18] C. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers", *In ACM SIGCOMM computer communication review*, Vol. 24, No. 4, 1994, pp. 234-244.

[19] S. Murthy and J. Garcia-Luna-Aceves, "An efficient routing protocol for wireless networks", *Mobile Networks and applications (MONET)*, Vol. 1, No. 2, 1996, pp. 183-197.

[20] C. Chiang, H. Wu, W. Liu and M. Gerla, "Routing in Clustered Multihop, Mobile Wireless Networks With Fading Channel", *proceedings of IEEE SICON*, Vol. 97, No. 4, 1997, pp. 197-211.

[21] S. Raut and H. Ambulgekar, "Proactive and Reactive Routing Protocols in Multihop Mobile Ad hoc Network", *International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE)*, Vol. 3, Issue 4, 2013, pp. 152-157.

[22] C. Perkins, E. Royer and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing", *Internet Engineering Task Force (IETF)*, RFC 3561, 2003.

[23] S. Raju, K. Runkana and J. Mungara, "ZRP versus AODV and DSR: A Comprehensive Study on ZRP Performance on MANETs", *International Conference on Computational Intelligence and Communication Networks*, 2010, pp. 194-199.

[24] M. Handley, H. Schulzrinne, E. Schooler and J. Rosenberg, "SIP: Session Initiation Protocol", *Internet Engineering Task Force (IETF)*, RFC 2543, 1999.

[25] W. Almobaideen, N. Kubba and A. Awajan, "FCSIP: Fuzzy and Cluster based SIP Protocol for MANET", *Eighth International Conference on Next Generation Mobile Applications, Services and Technologies (NGMAST)*, 2014, pp. 169-174.

[26] M. Alshamrani, H. Cruickshank and Z. Sun, "SIP Signaling Implementations and Performance Enhancement over MANET: A Survey", *International Journal of Advanced Computer Science and Applications (IJACSA)*, Vol. 7, No. 5, 2016, pp. 191-204.

[27] P. Sheeba and C. Vandana, "A New SIP-Based Application Layer Protocol for VoIP in MANET", International Journal of Engineering Research & Technology (IJERT), Vol. 4, Issue 1, 2015, pp. 733-737.

[28] L. Abdullah, I. Almomani, and A. Aburumman, "Secure Cluster-Based SIP Service over Ad hoc Networks", *IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT)*, 2013

[29] E. Gurumoorthi, A. Ayyasamy, M. Archana, and J. Barathy, "Performance Enhancement for QoS in VoIP Applications over MANET", *International Journal of Advances in Computer and Electronics Engineering*, Vol. 02, No. 05, 2017, pp. 47 – 54

[30] L. Andel, J. Kuthan, and D. Sisalem. "Distributed media server architecture for SIP using IP anycast", In *Proceedings of the 3rd International Conference on Principles, Systems, and Applications of IP Telecommunications*, 2009, pp. 5-15.

[31] P. Stuedi, M. Buhr, A. Remund, and G. Alonso, "SIPHoc: Efficient SIP Middleware for Ad Hoc Networks", *In Proceedings of the ACM/IFIP/USENIX International Conference on Middleware*, 2007, pp. 60-79.

[32] M. Jiang, J. Li,Y. C. Tay, "Cluster based routing protocol (CBRP) functional specification", *Internet Engineering Task Force (IETF)*, 1999.

[33] Y. Kumbharey, S. Shukla, and S.Chaturvedi, "Renovated Cluster Based Routing Protocol for MANET", *International Journal of Advanced Computer Research, Vol* 3, No.1, 2013, pp. 206-211.

[34] C. Solegaonkar and B. Prajapat, "An Overview on Cluster Head Election Techniques in Mobile Ad-Hoc Networks", *International Journal of Computer Applications*, Vol. 162, No. 9, 2017, pp. 10-12.

[35] S. Bhadoria, L. Nishad, and L. Tongue, "Review on Cluster-head Election Mechanisms for Clustering Based Routing in Mobile Ad-hoc Network", *International Journal of Scientific and Research Publications*, Vol. 4, Issue 7, 2014, pp. 1-3.

[36] M. Alinci, E. Spaho, A. Lala and V. Kolici, "Clustering algorithms in MANETs: a review", *Ninth International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS)*, 2015, pp. 330–335.

[37] A. Savyanavar and M. Borate, " Survey of Clustering Schemes in Mobile Ad hoc Networks", *International Journal of Science and Research (IJSR)*, Vol. 3, Issue 11, 2014, pp. 2407-2410.

[38] S. Leggio, J. Manner, A. Hulkkonen and K. Raatikainen, "Session Initiation Protocol Deployment in Ad-Hoc Networks: a Decentralized Approach", *In 2nd International Workshop on Wireless Ad-hoc Networks (IWWAN)*, Vol. 5, 2005.

[39] J. Veizades, E. Guttman, and C. Perkins, "Service Location Protocol*," Internet Engineering Task Force (IETF)*, RFC 2165, 1997.

[40] H. Chu and W. Chen, "Enabling SIP-Based Services in Ad Hoc Networks", *International Journal of Information Engineering (IJIE)*, Vol. 2, Issue 4, 2012, pp. 158-162.

[41] I. Mourtaji, M. Bouhorma, M. Benahmed, A. Bouhdir," A New Technique for Adapting SIP Protocol to Ad Hoc Networks: VNSIP (Virtual Network for SIP) Illustration and Evaluation of Performance", *International Journal of Computer Networks and Communications Security (IJCNCS)*, Vol. 1, No. 1, 2013, pp. 23-29.

[42] I. Mourtaji, M. Bouhorma, M. Benahmed, A. Bouhdir, "Performance Enhancement of VNSIP approach, using MCAC Algorithm", *International Journal of Computer Networks and Communications Security (IJCNCS)*, Vol. 1, No. 3, 2013, pp. 68-74.

[43] S. Nesargi, and R. Prakash. "MANETconf: Configuration of hosts in a mobile ad hoc network", *In INFOCOM Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, vol. 2, 2002, pp. 1059-1068.

[44] H.Jan and I. Ullah, "Assigning IP addresses to Mobile Adhoc Networks nodes by using Backup Source Node", *International Journal of Electrical & Computer Sciences IJECS-IJENS*, Vol. 11, No. 4, 2011, pp. 19-22.

[45] L.Villalba, J. Matesanz, A. Orozco, and J. Díaz, "Auto-Configuration Protocols in Mobile

Ad Hoc Networks", *Sensors*, Vol. 11, No. 4, 2011, pp. 3652-3666.

[46] H. Bandara, A. Jayasumana, and T. Illangasekare, "A Top-Down Clustering and Cluster-Tree-Based Routing Scheme for Wireless Sensor Networks", *International Journal of Distributed Sensor Networks*, Vol. 7, Issue 1, 2011.

[47] S. Shirke, V. Shah, T. Ruikar, and J. Abraham, "Cluster Based Hierarchical Addressing for Dynamic Source Routing", *In International Conference on Smart Trends for Information Technology and Computer Communications*, 2016, pp. 264-275.

[48] A. Yousef, H. Al-Mahdi, M. Kalil and A. Mitschele-Thiel, "LHA: Logical Hierarchical Addressing Protocol for Mobile Ad-hoc Networks", *In Proceedings of the 2nd ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks*, 2007, pp. 96-99.