

<http://ansinet.com/itj>

ITJ

ISSN 1812-5638

# INFORMATION TECHNOLOGY JOURNAL

**ANSI***net*

Asian Network for Scientific Information  
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

## Scalable and Effective Cluster Based Routing Algorithm Using Nodes' Location for Mobile Ad Hoc Networks

Y. Wang, T. Liang, X. Yang and D. Zhang

School of Electronics and Information Engineering, Xi'an Jiaotong University,  
Xi'an, ShaanXi, People's Republic of China

**Abstract:** In this study, we develop and analyze the Cluster Based Location-Aware Routing Protocol for Large Scale Heterogeneous Mobile Ad Hoc Networks (CBLARHM), a low-complexity routing algorithm for mobile ad hoc networks (MANET). CBLARHM runs on top of an adaptive cluster cover of the network, which can be created and maintained using, for instance, the weight-based distributed algorithm. The weighted clustering algorithm we proposed takes into consideration node degree difference, battery power, average link stability and average dependency probability of mobile nodes. The hierarchical structure stabilizes the end-to-end communication paths and improves the networks scalability such that the routing overhead does not become tremendous in large scale MANET. Furthermore, it is fascinating and important to investigate that how to control the total number of nodes involved in a routing establishment process so as to improve the network layer performance of MANET. CBLARHM is to use geographical location information provided by Global Position System (GPS) to assist routing. The location information of destination node is used to predict a smaller rectangle, isosceles triangle, or circle request zone, which is selected according to the relative location of the source and the destination, that covers the estimated region the destination may locates. Thus, instead of searching the route in the entire network blindly, CBLARHM confines the route searching space into a much smaller estimated range. Simulation results have shown that CBLARHM outperforms other protocols significantly in route setup time, routing overhead, mean delay and packet collision and simultaneously maintains low average end-to-end delay, high success delivery ratio, low control overhead, as well as low route discovery frequency.

**Key words:** Mobile ad hoc networks, routing, location, clustering, heterogeneous nodes

### INTRODUCTION

Scalable routing is one of the key challenges in designing and operating large scale Mobile Ad Hoc Networks (MANET). In order to ensure effective operation as the total number of nodes becomes large, the overhead of the employed routing algorithms should be low and independent of the total number of nodes in MANET. Developing routing protocols for MANET has been an extensive research area during the past few years and significant progress has been made in developing the algorithms and algorithm refinements to achieve scalable MANET routing. Among them, many earlier study (Blazevic *et al.*, 2005; Karp and Kung, 2002; Eriksson *et al.*, 2007; Iwata *et al.*, 2002; Jiageng *et al.*, 2005; KaiTen *et al.*, 2008; Kuruvila *et al.*, 2005; Wang and Olariu, 2004) are famous and classical. Recently, further approaches for routing on top of a cluster cover of a set of core nodes have been proposed (Jiang *et al.*, 1998;

Theoleyre and Valois, 2005; Wang and Olariu, 2004; Dixit *et al.*, 2005). CBRP (Jiang *et al.*, 1998) divides the nodes of the MANET into a number of overlapping or disjoint 2-hop-diameter clusters in a distributed manner. VSR (Theoleyre and Valois, 2005) is based on a virtual topology including both a backbone and clusters. The backbone collects control traffic and to reduce the overhead for route discoveries.

However, some key challenges remain in the development of scalable MANET routing algorithms. In particular, the existing MANET routing algorithms have been formally analyzed either:

- For the route discovery, a total elapsed time or total number of messages exchanged that depend on the overall network size, such as the total number of nodes in the MANET or the total diameter of the network (for instance, (Blazevic *et al.*, 2005; KaiTen *et al.*, 2008; Kuruvila *et al.*, 2005))

- Make restrictive assumptions about the overall network topology, such as limiting the network density (for instance, (Haas and Pearlman, 2001; Wang and Olariu, 2004; Ko and Vaidya, 2000; Karp and Kung, 2000; Woo and Singh, 2001))

For these reasons, the existing MANET routings are of limited use for large MANET consisting of a large number of nodes and having a large diameter. Typically, when wireless network size increases (beyond certain thresholds), common flat routing schemes become infeasible because of link and processing overhead. One way to solve the problem and to produce scalable and efficient solutions is hierarchical routing, which is based on the idea of organizing nodes in groups and then assigning nodes different functionalities inside and outside of a group. Both routing table size per node and the total number of update packet transmitted by the nodes are reduced because only part of the network (instead of the whole) are included, thus control overhead is reduced. However, the hierarchical routing always outperforms locally, when the routing path crosses long range, the number of nodes participated is still large. The cost of an independent node is reduced by using hierarchical routing mechanisms, but the total overhead of MANET may still be great. The efficient method is that the number of nodes involved in the routing process should be controlled by some aided techniques. One possibility direction is to use location information provided by GPS. Instead of searching the route in the entire network blindly, location-based routing protocol uses the location information of mobile nodes to confine the route searching space into a smaller estimated range. The smaller route searching space to be searched, the less routing overhead and the probability of broadcast storm problem will occur.

The main contribution of this study is to propose a scalable and effective routing protocol for a large scale MANET-the Cluster based Location-Aware Routing Protocol for Large Scale Heterogeneous MANET (CBLARHM). In contrast with these previous works, CBLARHM has two essential differences:

- CBLARHM aims to use the heterogeneous nodes to form the clusters to improve the scalability. In CBLARHM, a new heterogeneous clustering has been proposed as a basis of MANET routing. CBLARHM builds upon some specific properties of the underlying clustering structure. In particular, we present a cluster algorithm for heterogeneous nodes clustering in MANET. We propose a weight-based distributed clustering algorithm which takes into consideration the number of nodes a clusterhead

(CH) can handle reasonably (considering node MAC function), dependency probability, battery power and link stability of the mobile nodes. This algorithm assigns node-weights based on the suitability of nodes acting as CHs and the election of the CH is done on the basis of the largest weight among its neighbors. Since the entire network is partitioned into smaller logically separated clusters, the main advantage is that it is easy for a CH to keep track of mobile nodes in its radio coverage: nodes joining or leaving, cluster's topology and so on

- In CBLARHM, in order to reduce the routing overhead, some important and novel methods are adopted to improve the protocol's performance

Three scenarios of request zone, which are based on relative location between source node and destination node, are analyzed. The location information of destination node is utilized to predict an isosceles triangle, rectangle or circle request zone that reduces the coverage of route discovery space and covers the position of destination node. The smaller route discovery space reduces the total number of messages exchanged and the probability of collision.

Finally, to overcome the problem caused by the route hole, an increasing exclusive search approach is used to redo route discovery by a progressive increasing request zone when the initial route discovery failed. This method is also helpful to reduce routing overhead.

## CLUSTERING AS A BASIS FOR ROUTING

The CBLARHM protocol is implemented on the top of a clustering structure. In this section, we present the node clustering in detail.

**System model:** We consider a wireless system consisting of two types of nodes (type 0 and 1) which are distributed on a flat two-dimension field and let  $N$  denote the total number of nodes. Each node has a GPS receiver and the geographical position can be measured. We assume that nodes are uniquely identified, i.e., using the MAC addresses (ID). The nodes are with the capability to transmit with some different transmission ranges, which can, for instance, be achieved by power control (Ritchie *et al.*, 2006). The notations of the properties of the two types of nodes can be shown as Table 1. We do not assume any specific mobility model in our analysis,

Table 1: Notations and relations of two types of nodes

Node type	Type 0	Type 1	Relation
Initial energy	$e$	$E$	$e \leq E$
Max. speed	$v$	$V$	$v \geq V$
Max. radio range	$r$	$R$	$r < R$

although we conduct simulations for the Random Way-Point (RWP) mobility model. We only initially assume, as is reasonable and common, that the mobility of the nodes is on a time scale slower than the route discovery (Ritchie *et al.*, 2006).

**System parameters:** To obtain better hierarchy structure, here four main issues should be investigated: (1) the optimal ratio of the number of clusterheads (CHs) and the number of clustermembers (CMs), (2) the optimal hops between a CM and its related CH in same cluster, the purpose is to extend the scalability of MANET, (3) the dependency probability of a CM, which indicates the probability of a CM belongs to its cluste and (4) the link stability between two nodes, it means how stable a wireless link between two nodes is. Four system parameters should be introduced based on the solutions of these problems above: (1) the number of CHs  $n^*$ , (2) the optimal hops  $h^*$ , (3) the dependency probability  $P_d$  and (4) the link stability  $P_s$ .

**Optimal number of CHs  $n^*$ :** To solve the problem (1), the per node throughput can be considered as a key. According to Gupta and Kumar (2000), a node's transport capacity of a MANET with  $n$  mobile nodes is  $\Theta(W/\sqrt{n})$  bits/s. The authors also demonstrate that when the number of mobile nodes increases, the per node throughput declines rapidly. Thus, to improve the throughput, the efficient solution is to partition the mobile nodes into clusters, as shown in Fig. 1. All CHs are connected using powerful radio to form a higher-lever virtual backbone network (VBN). A CH has per node throughput in both its local cluster and VBN network since it is equipped with multiple radios. Assume that the total number of nodes in MANET is  $N$ , the number of CHs is  $n$  and that of CMs is  $N-n$ . In the optimal circumstance (Gupta and Kumar, 2000), if the network is partitioned equally, let  $W_1$  and  $W_2$  denote the channel bandwidth of the VBN and the local cluster, respectively, the per node throughput rate of a local cluster can be obtained as:  $S_{cluster} = \Theta(W_2/\sqrt{n})$ . Similarly, per node throughput rate of the VBN can be obtained as:  $S_{VBN} = \Theta(W_1/\sqrt{n})$ . Since,  $N$  is fixed, both  $S_{cluster}$  and  $S_{VBN}$  are only functions of  $n$ . We now investigate the traffic at a CH in detail. A CH has two interfaces, one for the local cluster and one for the backbone network. Traffic across clusters is switched at CHs from local cluster interfaces to backbone interfaces, or vice versa. Assume that network traffic is uniformly distributed. The portion of  $S_{cluster}$  of a CH used for traffic in/out to other clusters is  $(n-1)/n$ . For a CH, to avoid the channel congestion, this portion should be smaller or at most equal to  $S_{VBN}$ . So, we get:

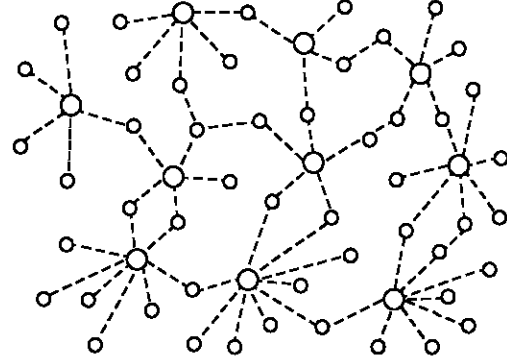


Fig. 1: General model of MANET

$$\frac{(n-1)}{n} S_{cluster} \leq S_{VBN} \quad (1)$$

Assume that the optimal number of CHs is  $n^*$ . Apparently, if  $n^*$  is used, the upper bound of per node throughput should be achieved. According to Gupta and Kumar (2000), the upper bounds of  $(n-1/n) S_{cluster}$  and  $S_{VBN}$  can be given as:  $\frac{n-1}{n} \sqrt{\frac{8}{\pi}} \frac{W_2}{\Delta} \sqrt{\frac{N}{n}}$  and  $\sqrt{\frac{8}{\pi}} \frac{W_1}{\Delta} \sqrt{n}$  respectively. Using  $n^*$  instead of  $n$ , the relationship between both bounds should be equal. Thus,

$$\frac{n^*-1}{n^*} \sqrt{\frac{8}{\pi}} \frac{W_2}{\Delta} \sqrt{\frac{N}{n^*}} = \sqrt{\frac{8}{\pi}} \frac{W_1}{\Delta} \sqrt{n^*} \quad (2)$$

According to the actual application, we choose the optimal number of CHs:

$$n^* = \frac{W_2 \sqrt{N} + \sqrt{W_2^2 N - 4 W_1 W_2 \sqrt{N}}}{2 W_1} \quad (3)$$

**Optimal hops  $h^*$ :** There are two different types of nodes in our analysis, because of the different functioning of the physical wireless devices, the relationship of the radio range, between a CM and a CH, is  $r < R$ . According to this setting, we have to propose the clustering algorithm to form  $h$ -hop clusters. To figure out the relationship between the number of CHs elected by our clusterhead election algorithm and the optimal number of CHs based on Eq. 3, we adjust the hops  $h$ , observe the change of the number of elected CHs, compare it with the optimal number  $n^*$ . According to actual application, we only calculate the  $n^*$  with  $W_1/W_2 = 1.5$  and  $W_1/W_2 = 2$ . As shown in Fig. 2, along with the increase  $h$ , the number of elected CHs decreases. Note that when  $W_1/W_2 = 2$  and the optimal hops  $h^* = 2$ , the value of the number of CHs elected by our clusterhead election is similar with the

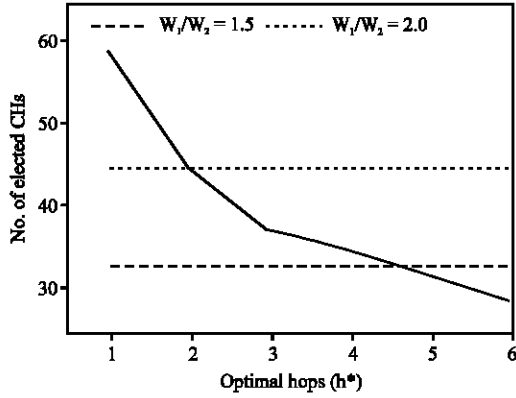


Fig. 2: No. of elected CHs as a function of h

value of optimal CHs obtained using Eq. 3. Thus, the results,  $h = 2$  and  $W_1/W_2 = 2$ , are our optimal hops  $h^*$  and system parameter in this study.

**Dependency probability ( $P_d$ ):** Because of the mobile characteristics of nodes, a CM may probably move out of the communication range of its CH. We refer the probability of a node  $i$  moving within  $j$ 's radio coverage as the dependency probability  $P_{d(i,j)}$  of  $i$  relative to  $j$ . Note that if the relative distance between  $i$  and  $j$ ,  $d(i,j)$ , gets smaller,  $i$  is closer to  $j$ . Based on the analysis above, we can define the equation of the dependency probability  $P_{d(i,j)}$  as:

$$P_{d(i,j)} = \frac{\pi(R^2 - d_{(i,j)}^2)}{\pi R^2} = 1 - \left(\frac{d_{(i,j)}}{R}\right)^2 \quad (4)$$

Note that when  $d(i, j) = r$ ,  $i$  only can receive  $j$ 's message and needs a relay node to communicate with  $j$ .

**Link stability ( $P_s$ ):** We define the link stability  $P_s$  between two nodes as a prediction of the time for which the nodes will be within communication range of each other. According to Frii's equation, the signal strength  $S$  is inversely proportional to the square of the relative distance  $d$  between two nodes:  $S \propto (1/d^2)$ . Thus, the  $P_s$  value should represent how long any particular node will offer a stable route. The approach we adopted to calculate  $P_s$  is described as follows:

- Each node in the ad hoc network broadcasts the message `Msg_Node_Hello` at a regular interval. Each node uses the received message to measure the signal strength of all the other nodes that are within its range.
- A node  $i$  calculates the  $P_s$  value of a neighboring  $j$  based on whether  $j$  is moving away or coming close according to the relative values of consecutive

messages, i.e., assume that the last two messages were received with signal strengths  $S_1$  and  $S_0$  (sorted by time,  $S_0$  is the signal strength measured by the latest message):

If  $S_0 \leq S_1$ , then it indicates that the two nodes are moving away. Thus, the  $P_s$  of  $i$  and  $j$  is calculated using Frii's equation:

$$P_s = \frac{R - d_1}{\Delta d} \Delta t = \frac{R - d_1}{d_1 - d_0} \Delta t = \frac{\sqrt{S_1/S_0} - 1}{1 - \sqrt{S_1/S_0}} \Delta t \quad (5)$$

where,  $d_1$  and  $d_0$  are the estimated relative distances corresponding to  $S_1$  and  $S_0$ ,  $\Delta t$  is the time interval between the two received signal and  $S_r$  is a pre-defined threshold, which is similar with the signal strength when the relative distance between two nodes is equal to  $r$ .

If  $S_0 > S_1$ , then it indicates that the two nodes are moving close. Thus, the  $P_s$  of  $i$  and  $j$  is calculated using Frii's equation:

$$P_s = \frac{R + d_1}{\Delta d} \Delta t = \frac{R + d_1}{d_1 - d_0} \Delta t = \frac{\sqrt{S_1/S_0} + 1}{\sqrt{S_1/S_0}} \Delta t \quad (6)$$

As mentioned above,  $d(i,j) \geq r$  means that the CM  $i$  can only receive its CH  $j$ 's message and it needs some relay CMs to communicate with CH  $j$ . Under this circumstance, assume that the multi-hop link from CM  $i$  to CH  $j$  is through relay node 1, node 2, ..., node  $n$ , the link stability  $P_{s(i,j)}$  is as follows:

$$P_{s(i,j)} = P_{s(i,1)} \cdot P_{s(1,2)} \cdots P_{s(n,j)} = \prod_{(a,b) \in \{(i,1), \dots, (n,j)\}} P_{s(a,b)} \quad (7)$$

**Definitions and notations:** The mobile ad hoc networks is formed by the two types of nodes and the links can be represented by an directed graph  $G = (V, E)$ , where a node set  $V = \{v\}$  and a node connectivity set  $E = \{e\}$ . The node set  $V$  gives both two types of nodes in the network and the connectivity set  $E$  denotes all the 1-hop links among the nodes in  $V$ . Because of the difference of physical radio devices of two types of nodes, we define that if the source and the destination of the link are the same type of nodes, the link is symmetric. For instance, if the link  $(m, n)$  between CM  $m$  and CM  $n$  exists, it means the link  $(n, m)$  definitely exists. Otherwise, the link is asymmetric. For instance, to CM  $i$  and CH  $j$ , if  $d_{(i,j)} = r$ , the links  $(i, j)$  and  $(j, i)$  exist, but if  $r < d_{(i,j)} = R$ , only link  $(j, i)$  exists. To be simple and clear, here are some definitions as follows:

**Definition 1:** A CM  $i$  uniquely joins a CH  $j$ , denoted as  $CM\ i \in CH\ j$ .

**Definition 2:** There are two CHs: CH  $j$  and CH  $k$  and a CM  $m$ . If  $m \in j$  and the links  $(m, j) \in E$ ,  $(j, m) \in E$ ,  $(k, j) \in E$ ,  $(j, k) \in E$  exist, the CM  $m$  is defined as Cross Node (CN) for CH  $j$  and CH  $k$ .

**Definition 3:** There are two CHs: CH  $j$  and CH  $k$  and two CMs: CM  $m$  and CM  $n$ . If  $(m \in j) \in (n \in k)$  and the links  $(m, j) \in E$ ,  $(j, m) \in E$ ,  $(n, m) \in E$ ,  $(m, n) \in E$ ,  $(k, j) \in E$ ,  $(j, k) \in E$  exist, the CM  $m$  and CM  $n$  are defined as Joint Nodes (JN) of CH  $j$  and CH  $k$ .

**Definition 4:** Gateway Node (GW) consists of two types: CN and JN.

**Definition 5:** There are two CHs: CH  $j$  and CH  $k$ , if CN or JN exists, CH  $j$  and CH  $k$  are defined as adjacent cluster head (ACH).

**Clustering procedure:** Clustering can be modeled as a graph partitioning problem with some added constraints. The critical issue is how to achieve an optimal VBN nodes deployment. The simplest way is to pre-assign VBN nodes and scatters them uniformly at initialization. However, such a static deployment has two main problems: (1) the type-1 nodes are moving, some type-1 nodes may move together while some areas lack of CHs and (2) CHs may fail or be out of battery or even destroyed, some new type-1 nodes should replace the defunct nodes. To solve these problems, our solution is to deploy redundant type-1 nodes (for instance, 110-120%  $n^*$ ). Since the number of type-1 nodes is more than strictly needed, only part of them become CHs. When one CH is destroyed or moves out of a certain area, a new CH will be selected from the redundant nodes.

As the search for more suitable type-1 nodes to be as CH for clustering continues, we propose a novel algorithm based on the use of a combined weight metric, that takes into account several system parameters. The cluster head election procedure is invoked at the time of system activation and also when the current CHs can be unable to cover all nodes. It needs to be emphasized that this procedure only involves all type-1 nodes.

**Cluster head election:** The procedure consists of seven steps as follows:

**Step 1:** Find the neighbour type 1 and 0 nodes of each type 1 node  $i$  (i.e., both types of nodes within its radio range) which defines its degree,  $Deg_i$ , as:

$$Deg_i = \sum_{j \in V, j \neq i} \{d_{(i,j)} < R_i\} \quad (8)$$

At the same time, construct its neighbour node set  $N(i)$  and the size of  $N(i)$  (i.e., the number of nodes in its neighbour node set) is  $K_{N(i)}$ .

**Step 2:** Compute the degree-difference  $\Delta d_i$  for each type 1 node:

$$\Delta d_i = \left| Deg_i - \frac{N}{n} \right| \quad (9)$$

**Step 3:** For every type 1 node, compute the average link stability  $P_s$ ,  $AP_s$ , with all its neighbours, as:

$$AP_s = \frac{1}{K_{N(i)}} \sum_{k=1}^{K_{N(i)}} P_{s(i,j)} \quad (j \in N(i)) \quad (10)$$

**Step 4:** Compute the average dependency probability  $P_d$ ,  $AP_d$ , with all its neighbours:

$$AP_d = \frac{1}{K_{N(i)}} \sum_{k=1}^{K_{N(i)}} P_d \quad (11)$$

**Step 5:** For every type 1 node, estimate the residual energy  $E_r$ .  $E_r$  implies how much battery power left. Apparently, the more energy left, the more suitable for a type 1 node to act as a CH is.

**Step 6:** Calculate the combined weight  $W_i$  for each type 1 node  $i$ , where

$$W_i = w_1 \Delta d_i + w_2 AP_s + w_3 AP_d + w_4 E_r \quad (12)$$

where,  $w_1$ ,  $w_2$ ,  $w_3$  and  $w_4$  are the weight factors for the corresponding parameters.

**Step 7:** Choose the node with the highest  $W_i$  as the CH. All the neighbours of the chosen CH are no longer allowed to participate in the election procedure.

**Cluster formation:** All nodes start in Cluster\_Undecided state. After generated the node weight, each type 1 node broadcasts its own combined weight in  $Msg\_Node\_Weight <ID, GPS(x, y), W>$  and the type 0 nodes send a message  $Msg\_Node\_Info <ID, GPS(x, y)>$  to its neighbors. Any type 1 node receives the message from its neighbors, stores the information in a special data structure called neighbor table NT (vector). Meanwhile, any type 1 node compares its own combined weight  $W$  with those of the neighbor table. If a type 1 node has a highest value of  $W$  amongst all its neighbors, it assumes the status of Cluster\_Head\_Candidate and then declares itself to be as a CH candidate node; otherwise it declares

itself to be a Cluster\_Member and then waits to join a CH. If two neighboring CH candidate nodes in Cluster undecided state have the same value of W, we resort to comparison of average link stability. The node, whose  $AP_s$  is greater, is decided to be as the CH and changes its status to be Cluster\_Head; the loser has to change its status to be Cluster\_Member. Furthermore, because in a mobile scenario, if two type 1 nodes with status Cluster\_Head move into each other's communication range and they are in contention of retaining the Cluster\_Head position, the Event\_Retain\_CH occurrence is deferred for a CH\_Contention\_Interval (i.e., 4ACK\_Timeout\_Interval) to allow for the incidental contacts between passing nodes. If the type 1 nodes are in communication range of each other even after the CH\_Contention\_Interval timer has expired, the Event\_Retain\_CH occurs; the type 1 node with the greater  $AP_s$  keeps the status of Cluster\_Head and the other changes its status to be Cluster\_Member.

To connect the CHs, every CH sends a message Msg\_Connect\_CH to its CMs. When relaying the message, the node, which receives and then relays the message, adds its information  $\langle ID, GPS(x,y) \rangle$ . The CM, if and only if whose role is GW, receives and relays the message to its next hop; otherwise, the message is dropped. If the next hop is CH, it receives the message, stops relaying the message and stores it in its neighbor CH table NHT (vector).

### CBLARHM PROTOCOL

Here, we describe our Cluster based Location-Aware Routing Protocol for Large Scale Heterogeneous MANET (CBLARHM) which runs on top of a cluster cover of the network. Firstly, some necessary definitions are given, including expected zone and request zone. Secondly, three scenarios of request zone are analyzed. Thirdly, route discovery, hole problem and route recovery are discussed.

**Propagation of location information:** Initially, in mobile ad hoc networks, a node may not know the physical location (either current or old) of other nodes. However, similar with the LAR (Ko and Vaidya, 2000), we consider that, as time progresses, each node can get location information for many nodes either as a result of its own route discovery or as a result of message forwarding for another node's route discovery. For instance, if node m includes its current location in the route request message and if node n includes its current location in the route reply message, then each node receiving these messages can know the current locations of not only the nodes m and n,

but also the relay nodes. Besides, once a node receives these messages, the location information of the nodes participated in the routing process can be updated. Generally, location information may be propagated by piggy-backing it on any packet.

**Expected zone:** In the route discovery procedure, the source S uses the location information of the destination D to estimate the region that the destination expects to appear, the region is called as expected zone (Ko and Vaidya, 2000). We extend the definition of expected zone in LAR, because many literatures have proved that the method proposed in LAR can not calculate the expected zone exactly (Dixit *et al.*, 2005; Iwata *et al.*, 2002). However, when the route request message arrives the original location of D, some time passed, this time interval is called  $\Delta t$ . As shown in Fig. 3, in order to calculate a more exact expected zone, we must take the time interval  $\Delta t$  into account. There are two scenarios: (1) the routing path from S to D has been established. S knows the transmission time of a message from D to S, so the  $\Delta t$  can be estimated as the transmission time from D to S and (2) the routing path from S to D is not established. In this scenario, a challenge is how to determine the value of  $\Delta t$ . For simplicity,  $\Delta t$  can be set to half of the round trip time between S and D. Another more precise and complex method to get the value of  $\Delta t$  is as follows: when S received a packet from D, S adds the location information of D and the transmission time of this packet from D to S into its routing table. If S needs to calculate an expected zone for D, we can let  $\Delta t$  as the transmission time from D to S that is recorded in the routing table of S. Particularly, if any node also propagates to other nodes its average velocity (or some other measure of velocity). In this case, S knows the D's location  $GPS(x_D, y_D)$ , its location  $GPS(x_S, y_S)$ , its velocity  $v_S$  and D's velocity  $v_D$ . The  $\Delta t$  can be given as:  $\Delta t = \sqrt{(x_S - x_D)^2 + (y_S - y_D)^2} / v_D$ . Finally, the radius of estimated expected zone is  $v_D [(t_1 - t_0) + \Delta t]$ .

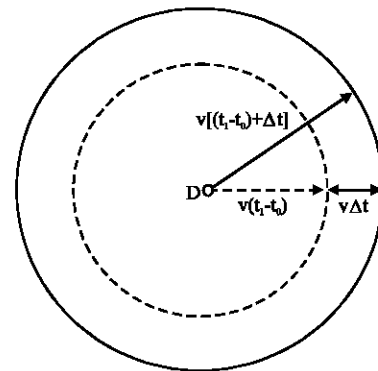


Fig. 3: The expected zone

**Request zone:** Instead of searching the route in the entire network blindly, our CBLARHM confines the route searching space into a smaller estimated region, which is defined as request zone. A node forwards a route request only if it belongs to the request zone. To increase the probability that the route request will reach the destination, the request zone should not only include the expected zone but also other region around it and the routing path. Thus, three scenarios of request zone should be considered: (1) S is outside of expected zone and S and D are in same cluster; (2) S is outside of expected zone and S and D are in different clusters; (3) S is within the expected zone. Next, we discuss three scenarios in detail.

**Scenario 1:** If S is outside of the expected zone and S and D are in same cluster, the request zone is defined as an isosceles triangle, named RZI, which includes the current location of S and the estimated expected zone EZ. As shown in Fig. 4, the area of RZI, whose corners are S, A and B, can be calculated as follows:

$$S_{\Delta ASB} = (d+r)^2 \tan \alpha$$

$$= \frac{\{v_D[(t_1 - t_0) + \Delta t]\} \left\{ \sqrt{(x_S - x_D)^2 + (y_S - y_D)^2} + v_D[(t_1 - t_0) + \Delta t] \right\}^2}{(x_S - x_D)^2 + (y_S - y_D)^2 - v_D^2[(t_1 - t_0) + \Delta t]^2} \quad (13)$$

There are two advantages: (1) RZI restrains the route request message to flood in a narrower space. It means that the request message is sent to D from S as directly as it can. Apparently, it provides a higher chance for S to select a short routing path to D. It indicates that the delay of route is small; (2) the area of RZI is much smaller than that of LAR. It means that RZI confines the route search to a smaller space. It indicates that the overhead of route is smaller than that of LAR.

**Scenario 2:** If S is outside of the expected zone and S and D are in different clusters, the request zone is defined as a rectangle, named RZII. Because of the heterogeneous nodes in the MANET, they have different physical properties. The VBN links among CHs provide short cut and additional bandwidth. Routing algorithm should be able to utilize them for remote destination nodes. Intuitively, the routing path passes more CHs, the total number of nodes the routing path through is less; thus, the delay is smaller and the overhead is lower. The critical issue in Scenario 2 is that the restricted region should provide a higher chance to let the request message route through the CHs as many as possible.

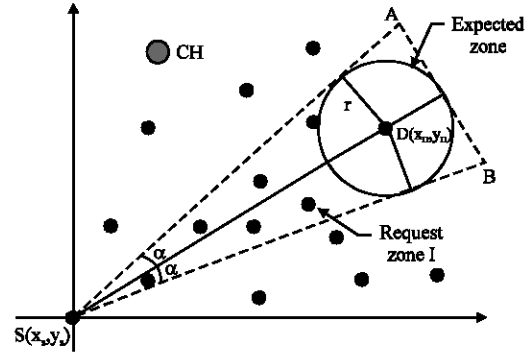


Fig. 4: Scenario of the request zone I

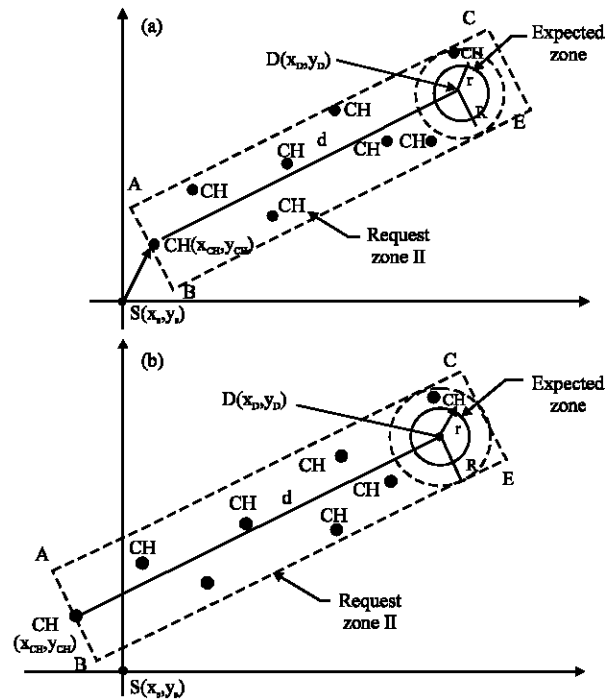


Fig. 5: Scenario of the request zone II, (a) CH is closer to D and (b) S is closer to D

As shown in Fig. 5, different from the request zone RZI started by the source node S, RZII is started by the CH which S belongs to. The request message is forwarded from S to its CH; CH checks it and starts the RZII procedure. Thus, RZII includes the current location of CH and the estimated expected zone EZ.

Inspecting Fig. 5, we found that the RZII corners are A, B, C and E. The area of RZII can be calculated as follows:

$$S_{WABCE} = 2R \cdot (d+R)$$

$$= 2R \cdot \left( \sqrt{(x_{CH} - x_D)^2 + (y_{CH} - y_D)^2} + R \right) \quad (14)$$



where,  $R$  is the maximum radio radius of a CH.

There are two reasons we choose  $R$  other than  $r$ . Firstly, the distance between CH and CM is  $R$  at most. In the hierarchical VBN, if  $D$  is a CM, we can route the request message to the CH that  $D$  belongs to. The CH manages  $D$  and communicates with it easier. Secondly, the routing path passes the CHs as many as it can. The larger the area of the request zone is, the more chance the routing path passing more CHs. Each CH has more energy and performs much longer range transmissions to next hop node. Thus, the total number of nodes the routing path through is less, the delay is smaller and the overhead is lower.

**Scenario 3:** If  $S$  is within the expected zone, the request zone is defined as a circle, named RZIII, which is equal to the expected zone, shown in Fig. 6.  $S$  locally sends the request message in the RZIII. In particular, if and only if when the Scenario 3 is used,  $S$  and  $D$  are in same cluster or adjacent clusters, as shown in Fig. 6,  $S$  and  $D$  can ignore the hierarchy temporarily. It means that  $S$  can send the request message in the RZIII without being passed through any CH, it is quite obvious that this method can improve the algorithm efficiency.

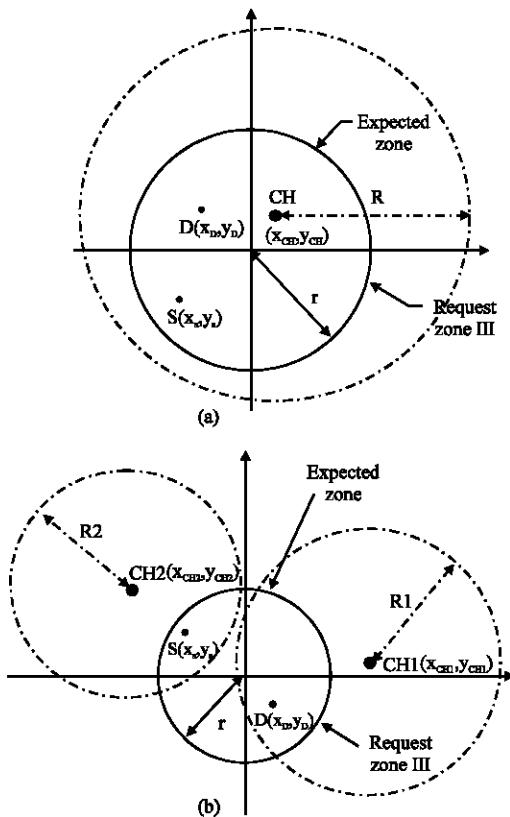


Fig. 6: Scenario of the request zone III, (a)  $S$  and  $D$  in a same cluster and (b)  $S$  and  $D$  in adjacent clusters

**Hole problem:** Because of the narrow space of each request zone and the mobile characteristics of nodes, if there are holes in the request zone, the route discoveries are influenced and likely to be repeated many times, which in turn increases the routing overhead and extends the delay of routing path. To overcome the problem caused by the hole, a hole detection method is proposed.

In scenario 1,  $S$  forwards the request message to some relay node  $i$ ,  $i$  checks if there are next hop neighbor nodes locate within in the RZI by using the neighbor nodes' location information that are recorded in its NT. If there is no neighbor node suitable for being as the next hop,  $i$  returns the `Msg_Node_RErr` to  $S$ , which includes the location information of neighbor node  $j$  which node  $i$  consider is suitable for being as the next hop. After having received the message,  $S$  will increase the angle  $\alpha$  to  $\alpha'$  and recalculate RZI. As shown in Fig. 7a, the line through  $S(x_s, y_s)$  and  $j(x_j, y_j)$  is given by:

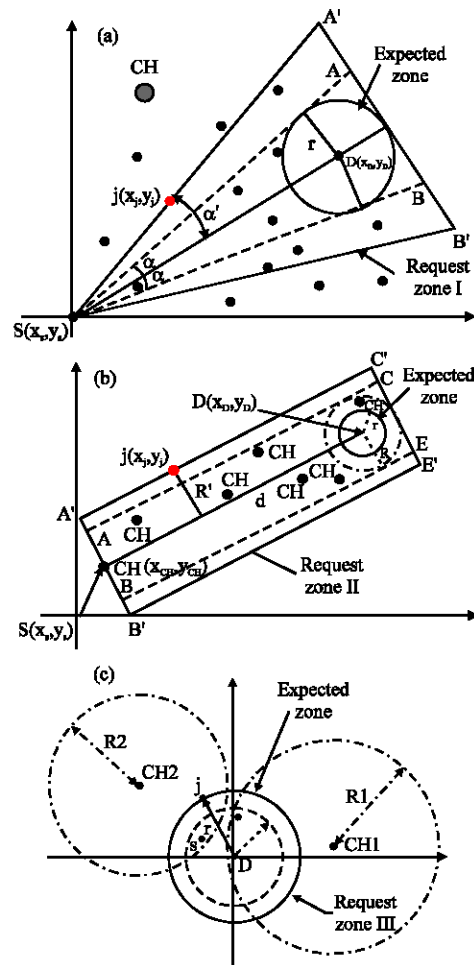


Fig. 7: New request zone, (a) New RZI and (b) New RZII and (c) New RZIII

$$Y = \frac{y_s - y_j}{x_s - x_j} (X - x_j) + y_j \quad (15)$$

Similarly, line through S ( $x_s, y_s$ ) and D ( $x_d, y_d$ ) is given by:

$$Y = \frac{y_s - y_d}{x_s - x_d} (X - x_d) + y_d \quad (16)$$

Thus, the new  $\alpha'$  can be calculated as follows:

$$\alpha' = \arctan \frac{\left( \frac{y_s - y_j}{x_s - x_j} \right) - \left( \frac{y_s - y_d}{x_s - x_d} \right)}{1 + \left( \frac{y_s - y_j}{x_s - x_j} \right) \left( \frac{y_s - y_d}{x_s - x_d} \right)} \quad (17)$$

In scenarios 2 and 3, the method we proposed is similar. The idea is to enlarge the coverage of request zone. In the scenario 2, after having received the message `Msg_Node_RReq`, S will extend the line segment CHA to CHA', as shown in Fig. 7b. The new  $R'$  can be calculated as follows:

$$R' = \frac{|(x_d - x_{ch})x_j + (y_{ch} - y_d)y_j + (x_{ch}y_d - y_{ch}x_d)|}{\sqrt{(x_d - x_{ch})^2 + (y_d - y_{ch})^2}} \quad (18)$$

In scenario 3, S will extend the radius of RZIII. The new radius  $r'$  is the distance from j to D, as shown in Fig. 7c.

**Route discovery:** When the source S wants to transmit data packets to the destination D, it performs a series of operations to estimate the expected zone and request zone and then it establishes the routing path.

The route discovery procedure consists of seven steps:

**Step 1:** S calculates an expected zone by the approach we described in (section expect zone), while it uses the basic information of the destination node D  $\langle CH\_ID(D), ID(D), GPS(x_d, y_d) \rangle$ . Meanwhile, the distance between S and D  $dist(S, D)$  can be obtained.

**Step 2:** S judges the scenario the request zone belongs to:

```
1.procedure Judge_scenario
2.begin
3.  if (dist(S,D) <= vD[(t1-t0)+Δt])
4.      then scenario(3) is initiated;
5.          return RZIII;
```

```
6.  else if (CH_ID(S) == CH_ID(D))
7.      then scenario(2) is initiated;
8.          return RZII;
9.  else if scenario(1) is initiated;
10.      return RZI;
11.end
```

**Step 3:** S defines a request zone to include the expected zone according to the analysis above.

**Step 4:** S sends a route request message, named `Msg_Node_RReq`, that includes the information of the request zone and the basic information of D, whose options are  $\langle type, sour, dest, RZ(X), pathlist, h, routed \rangle$ . S sets  $type=request$ ,  $sour = ID(S)$ ,  $dest = ID(D)$ ,  $h = 0$ ,  $routed = 0$ ,  $pathlist = null$  and the value of X in  $RZ(X)$  can be decided based on procedure `Judge_scenario` in step 2.

**Step 5:** After received the `Msg_Node_RReq`, intermediate node i invokes the procedures, until the `Msg_Node_RReq` reaches to D,  $routed = 1$ . The pseudo-code of each procedure is as below:

```
1.procedure Establish_path
2.begin
3.  Receive(Msg_Node_RReq);
4.  if (i ∈ RZ)
5.      then Discard (Msg_Node_RReq);
6.      exit(0);
7.  else if (i ∈ RZ(X))
8.      then case X of
9.          1: call Routing_RZI; break;
10.         2: call Routing_RZII; break;
11.         3: call Routing_RZIII; break;
12.end
13.procedure Routing_RZI
14.begin
15.  if (i == CH)
16.      if (D communicates with i directly)
17.          then add i to pathlist; h = h + 1;
18.          else then select next hop j;
19.              call Judge(j);
20.  if (i == CM)
21.      if (i checks whether D is its
22.          adjacent CM)
23.          then add i to pathlist; h = h + 1; routed = 1;
24.      else then select the next hop j;
25.          call Judge(j);
26.procedure Routing_RZII
27.begin
```

```

28. if (i == CH)
29.   if ((D is its CM) and (D
communicates with i directly))
30.     then add i to pathlist; h = h
+ 1; routed =1;
31.   if ((D is its CM) and (D cannot
communicates with i directly))
32.     then select next hop j;
33.     call Judge(j);
34.   if (D is not i's CM )
35.     then select next hop m from
i's N(i) and CH is preferred;
36.     call Judge(m);
37.   if (i == CM)
38.     if (i checks whether D is its
adjacent node)
39.       then add i to pathlist; h =
h + 1; routed =1;
40.     else then select next hop n from
i's N(i) and CH is preferred;
41.     call Judge(n);
42.end
43.procedure Routing_RZIII
44.begin
45.  select next hop j from i's N(i);
46.  call Judge(j);
47.end
48.procedure Judge(node j)
49.begin
50.  if (j ∈ RZ)
51.    then return Msg_Node_RErr to S;
52.  else (j ∈ RZ)
53.    then send Msg_Node_RReq to j;
59.    add node i itself to pathlist;
h = h + 1;
55.end

```

In the pseudo-code, we assume that the next hop node is always closer to D. Each intermediate node *i* invokes procedure Establish\_path to check whether it is able to be involved in this routing. Firstly, extracted from Msg\_Node\_RReq, node *i* estimates the type of scenario which request zone will obtain. Then *i* performs the next procedure. Because of three types of request zone scenarios, strategy *i* employed need consider the relation of relative location of S, D and EZ. In RZI, S, *i* and D are in same cluster and the challenge is the link may be asymmetric because of the radio radius difference between CH and CM. Thus, if *i* is CH, it query NT (vector) to check whether the link from D to *i* is symmetric (line 16~19). When *i* is CM, *i* searches its Table\_Node\_Neighbor (vector), checks whether D is its

neighbor node (line 20~24). In RZII, S and D are in different clusters. The main problem is that how to guarantee the strategies *i* employed can pass as many CHs as possible. When *i* chooses the next hop node, *i* prefers CH node in its neighbor node set N(*i*) (line 35,40). In RZIII, request zone is equal to expected zone initially. The challenge is that how to improve the efficiency. The strategy *i* employed is that S and D can ignore the hierarchy temporarily and S chooses the next hop in its NT.

**Step 6:** When Msg\_Node\_RReq is received by the destination D, it unicasts a route reply message along the reverse direction of the route that is recorded in the request packet to the source S. If D has received multiple pieces of route request message, D chooses the one with the least *h* to reply. The route reply message contains <CH\_ID(D), ID(D), GPS( $x_D, y_D$ ), pathlist>, which is named Msg\_Node\_RRly.

**Step 7:** S waits for receiving the route reply message from D. After that, the routing path from S to D is established.

**Route recovery:** If a route failure is detected by an intermediate node in the routing path to the destination D, or the source S does not receive any reply message, the route must be recovered as soon as possible. If the route failure is detected by an intermediate node, there are two methods to repair the route. The first method is to initiate a route discovery process by the broken node, called local search, to repair the broken path. This method is investigated in LAR (Ko and Vaidya, 2000) and it can reduce the overhead of route recovery as well as the latency of route rediscovery. If the local search method fails, the second method should be employed. The second method is the node detected the route failure sends back a route error message Msg\_Route\_Fail to inform the source a route failure has occurred. After having received the message, the source re-initiates a route discovery to search for a new routing path.

CBLARHM may be used in combination with the time-to-live (TTL) optimization. By setting the TTL to some reasonable number, the source can bound the number of hops the request will travel. Therefore, even if a node exists within the request zone defined by the CBLARHM, it will drop the message when it is over TTL hops away from the source S. Thus, in the case when S can receive any reply message within a suitable timeout period, CBLARHM allows S to initiate a new route discovery with an expanded request zone, which has been discussed in (Section Hole Problem).

## PERFORMANCE EVALUATION

Here, we present simulation results to illustrate the performance of CBLARHM protocol. The simulator is implemented within Global Mobile Simulation (GloMoSim) library by C++ language (Gajaj *et al.*, 1998). The GloMoSim library is a scalable simulation environment for mobile wireless network using parallel discrete-event simulation capability provided by PARSEC (Bagrodia *et al.*, 1998). We examine two aspects of the CBLARHM in simulations, namely, (1) the topology structure performance of clustering algorithm, which is as a basis of the CBLARHM, (2) the comparison of the network layer performance of the well-known CBRP, VSR, LAR and TZRP with CBLARHM.

In this simulation, all network nodes are located in a physical area of size  $1000 \times 1000 \text{ m}^2$  to simulate actual mobile ad hoc networks. The network size is in the range of [200, 300, 400, 500, 600, 700, 800] nodes that were generated according to a uniform distribution. In the simulations, we set  $e = 25 \text{ J}$ ,  $E = 35 \text{ J}$ ,  $v[0, 35]$ ,  $v[0, 30]$ ,  $r = 40 \text{ m}$ ,  $R = 100 \text{ m}$ ,  $W_1 = 2 \text{ Mb sec}^{-1}$ ,  $W_2 = 1 \text{ Mb sec}^{-1}$ . We conduct simulations for the RWP mobility models; the pause time is fixed to 30 sec. For random waypoint, a node randomly selects a destination from the physical terrain and then it moves in the direction of the destination in a speed uniformly chosen between the minimum and maximum roaming speed. After it reaches its destination, the node stays there for a specified pause time period. The propagation path loss model used is the TWO-RAY model that uses free space path loss (2.0, 0.0) for near sight and plane earth path loss (4.0, 0.0) for far sight. The antenna height is hard-coded in the model (1.5 m). We assume that different frequency bands for the intra-cluster communication inside the individual clusters and the inter-cluster communication among adjacent cluster leaders. This simulation model considers the Distributed Coordination Function (DCF) of 802.11, which employs carrier sense multiple access with collision avoidance (CSMA/CA). We do not employ request-to-send/clear-to-send (RTS/CTS) reservations for the RREQ packets to avoid the reservation overhead for these short packets.

The simulation time of each round lasts for 1000 sec. Each simulation result was obtained from an average of the all simulation statistics. In each round, there are four application connections. The traffic generators used by the four application connections are constant bit rate (CBR). The CBR simulates a constant bit rate traffic generator. The generators initiated the first packet (i.e., start time) in different time and sent a 512 bytes packet each time.

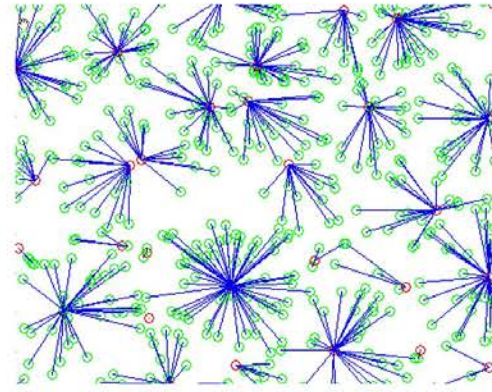


Fig. 8: The local output of simulation results

Table 2: Clustering algorithm efficiency for different N

N	n-n*	CS	HC	OH	OM	OHR	OMR
200	1.77	9.714	1.008	3.201	2.564	8.101	5.211
300	1.74	9.784	1.020	3.214	2.588	8.141	5.267
400	1.41	11.237	1.215	3.239	2.641	8.125	5.199
500	1.09	11.194	1.228	3.199	2.687	8.247	5.254
600	0.52	12.983	1.346	3.218	2.741	8.273	5.248
700	0.45	14.812	1.551	3.227	2.894	8.228	5.249

### Topology structure performance of CBLARHM:

Figure 8 shows that the local output of one of the simulation results of our algorithm at some time. As shown in Fig. 8, the distribution of CHs is relatively uniform and the size of cluster is even. As mentioned earlier, the size of cluster needs to be small enough, but because of the dynamic movements of the mobile nodes, the perfect scenario can not be obtained.

Table 2 shows the performance results according to six usual clustering metrics: (1) average Cluster Size (CS) is the average number of CMs included in each cluster, (2) average Hop Counts (HC) is the average hop counts of each CM to its CH, (3) average communication overhead of a CH per cluster formation (OH) is the average number of messages sent by each CH in cluster formation, (4) average communication overhead of a CM per Cluster Formation (OM) is the average number of messages sent by each CM in cluster formation, (5) average communication overhead of a CH per round (OHR) is the average number of messages sent by each CH in entire clustering process per round and (6) average communication overhead of a CM per round (OMR) is the average number of messages sent by each CM in entire clustering process per round. These metrics permit a more accurate evaluation of the quality of the obtained clustering structure are briefly observed. CS gauges the load imposed on CHs. Table 2 shows that the number difference between  $n$  and  $n^*$  decreases as the total number of node increases. That means that the cluster head election and cluster formation algorithm is effective

and suitable for large-scale network. HC permits better estimating the hierarchy structure of cluster. Note that as the total number of nodes increases, the variation of HC increases slowly. It demonstrates that the distribution of CHs is uniform and the size of cluster is even. OH and OM measure the overhead of communications between CMs and CHs in cluster formation, OHR and OMR indicate the overhead of communication between CMs, temporary nodes and CHs in entire clustering process per round. The value of four parameters is stable, OH is stable at around 3.2, OM is at around 2.65, OHR is at around 8.2, OMR is at around 5.2. That means our proposed algorithm is efficient in controlling the communication overhead and guarantees that various overheads do not increase acutely.

**Comparison with four protocols:** CBLARHM is compared with some famous and classical protocols, such as CBRP (Jiang *et al.*, 1998), VSR (Theoleyre and Valois, 2005), LAR (Ko and Vaidya, 2000) and TZRP (Wang and Olariu, 2004). Four performance metrics are introduced to evaluate the routing performance of CBLARHM:

- **Average end-to-end delay:** The end-to-end delay is averaged over all surviving data packets from the source S to the destination D.
- **Success delivery ratio:** Ratio of data packets delivered to the destination D to those generated by the source S.
- **Route discovery frequency:** The total number of route discoveries initiated per second.
- **Control overhead:** The total number of routing control packets normalized by the total number of received data packets.

From Fig. 9, CBRP shows fast increase in packet end-to-end delay. The reason is that when there is a large amount of control packets contenting for channel usage, the data packets have to back off a lot for a free slot. VSR usually has large routing packets but fewer control packets than CBRP, so, the delay is shorter than VSR. The packet end-to-end delay in LAR increases slightly because the location of a node is constantly updated via location update messages sent by the moving node and therefore changes in the topology have little effect on the delay. TZRP uses two zones to limit the nodes involved in the route discovery and reduces the control packets. CBLARHM performs much better than other four protocols in more stressful (i.e., larger number of nodes, more load), that is greatly contributed to the establishment of the request zone and three routing strategies of request zone we proposed.

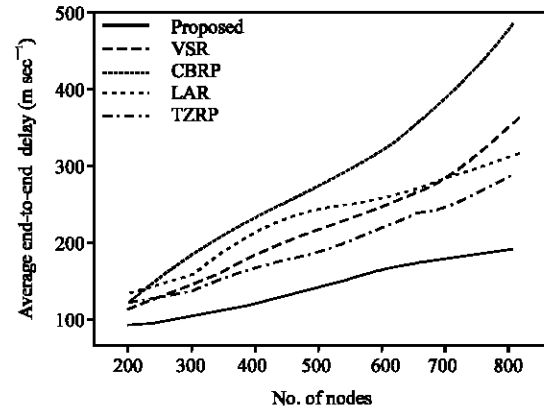


Fig. 9: Average end-to-end delay with varying the number of nodes

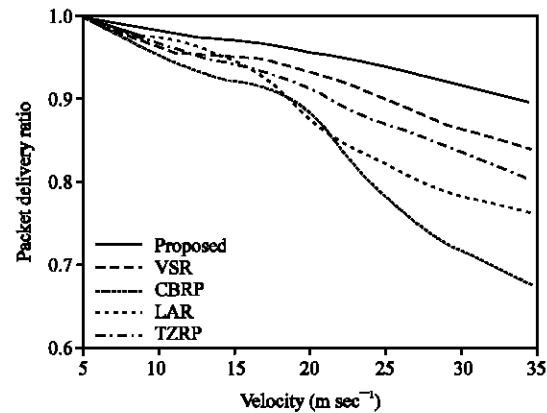


Fig. 10: Success delivery ratio with varying the maximum of velocity of nodes

Figure 10 shows the results of success delivery ratio for CBLARHM, CBRP, VSR, LAR and TZRP. It illustrates that CBLARHM outperforms at any mobility speed, especially exhibited higher performance at higher speed. From Fig. 10, when the mobility speed = 30 m sec<sup>-1</sup>, because CBLARHM has two methods to recovery the failure path, it lost fewer packets than CBRP, VSR, LAR and TZRP: 41.52, 37.48, 19.78 and 24.51%, respectively. The results demonstrate that CBLARHM may provide efficient fault tolerance in the sense of faster and efficient recovery from route failures in dynamic networks.

Figure 11 shows the results of discovery frequency performance. CBLARHM needs less discovery times to maintain these routing paths. CBRP is a simple path routing protocol based on cluster, so the source must broadcast a lot of discovery packets to recover the broken path. VSR cannot use the local search mechanism to repair the broken path, but always waits the source node's response. Thus, VSR also has more discovery times than



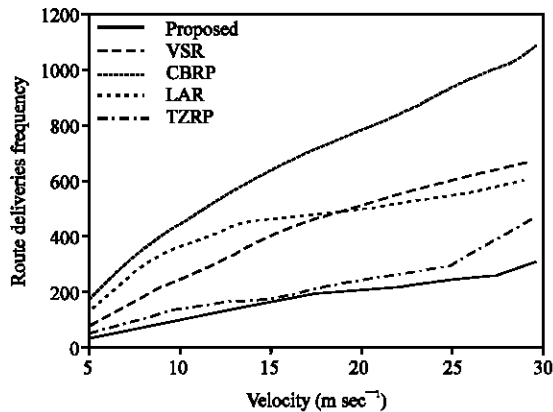


Fig. 11: Route delivery frequency with varying the maximum of velocity of nodes

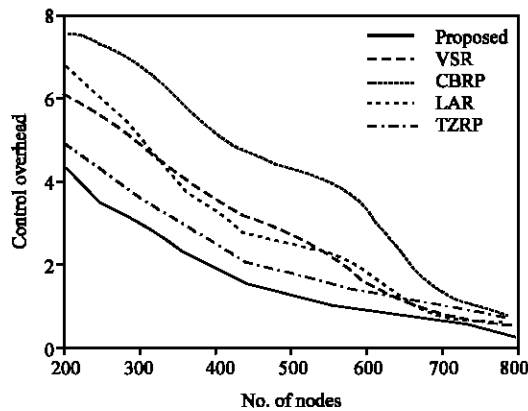


Fig. 12: Control overhead with varying number of nodes

that of CBLARHM. Both LAR and TZRP use locality information to reduce the route discovery frequency. LAR relies on a location update mechanism that maintains approximate location information for all nodes in a distributed fashion. While nodes moving, the approximate location information is constantly updated. TZRP uses Crisp zone for proactive routing and efficient broadcasting and a Fuzzy zone for heuristic routing using imprecise locality information. The results demonstrate a desirable property of CBLARHM that the routes still remain with high probability even at high rates of mobility. It is interesting to observe that the effects of the parameters in the clustering algorithm on this metric.

Figure 12 shows the results of control overhead as a function of the node density in MANET. The control overhead includes that route request packet and route reply packet for a node involved in the routing process. The total numbers of overheads per node among five protocols increase when the number of nodes increases. With a higher node density of MANET, the performance

of each protocol remains unaffected. The simulation results show that the control overhead of CBLARHM is lower than that of CBRP, VSR, LAR and TZRP, especially when the number of nodes increases large enough. By comparison, we can notice from Fig. 12 that the larger the size of the network is, the lower the control overhead of CBLARHM is relative to the other four protocols.

## CONCLUSION

In this study, we have developed and analyzed a novel Cluster based Location-Aware Routing Protocol for Large Scale Heterogeneous MANET (CBLARHM) for the design and operation of the large scale wireless mobile ad hoc networks. MANET is dynamic in nature due to the mobility of nodes. The association and dissociation of nodes to and from clusters perturb the stability of the network topology and hence a reconfiguration of the system is often unavoidable. However, it is vital to keep the topology stable as long as possible. These clusterheads, forming a virtual backbone in the network, determine the network's topology and stability. A weight-based clustering algorithm is used by CBLARHM to establish a cluster cover of the networks and reduce routing control overhead and improve the networks scalability. This algorithm takes into consideration the node degree, mobility, relative distance, battery power and link stability of mobile nodes. Moreover, using the location information of mobile nodes to assist routing can confine the route searching space into a smaller estimated range. The mechanism we adopted is to use geographical location information provided by GPS to assist routing. The location information of destination node is used to predict a smaller rectangle, isosceles triangle, or circle request zone, which is selected according to the relative location of the source and the destination, that covers the position of destination in the past. Instead of searching the route in the entire network blindly, CBLARHM limits the search for a routing path to the so-called request zone; it is obvious that the smaller route discovery space reduces the traffic of route request and the probability of collision. Simulation results have shown that CBLARHM outperforms other protocols significantly in route setup time, mean delay, routing overhead and collision and simultaneously maintains low average end-to-end delay, high success delivery ratio and low route discovery frequency.

## ACKNOWLEDGMENTS

The authors would like to thank China Next Generation Internet (CNGI 04-1-11D) and National High-Tech and Development Plan of China, under grant 863-2006AA01Z210, for funding this project.

## REFERENCES

- Bagrodia, R., R. Meyer, M. Takai, Y. Chen, X. Zeng, J. Martin, B. Park and H. Song, 1998. Parsec: A parallel simulation environment for complex systems. *Computer*, 31: 77-85.
- Blazevic, L., J.Y. Le Boudec and S. Giordano, 2005. A location-based routing method for mobile ad hoc networks. *IEEE Trans. Mobile Comput.*, 4: 97-110.
- Dixit, S., E. Yanmaz and O.K. Tonguz, 2005. On the design of self-organized cellular wireless networks. *IEEE Commun. Mag.*, 43: 86-93.
- Eriksson, J., M. Faloutsos and S.V. Krishnamurthy, 2007. DART: Dynamic Address Routing for Scalable Ad Hoc and Mesh Networks. *IEEE ACM Trans. Network*, 15: 119-132.
- Gajaj, L., M. Takai, K. Tang, R. bagrodia and M. Gerla, 1998. GloMoSim: A library for parallel simulation of large-scale wireless networks. *Proceedings of the 12th workshop on Parallel and distributed simulation*, ACM, 1998 Alberta, Canada, pp: 154-161.
- Gupta, P. and P.R. Kumar, 2000. The capacity of wireless networks. *IEEE Trans. Inform. Theory*, IT, 46: 388-404.
- Haas, Z.J. and M.R. Pearlman, 2001. The Performance of Query Control Schemes for the Zone Routing Protocol. *IEEE. ACM Trans. Network*, 9: 427-438.
- Iwata, A., C. Ching-Chuan, P. Guangyu, M. Gerla and C. Tsu-Wei, 2002. Scalable routing strategies for ad hoc wireless networks. *IEEE J. Sel. Area Commun.*, 17: 1369-1379.
- Jiageng, L., D. Cordes and Z. Jingyuan, 2005. Power-aware routing protocols in ad hoc wireless networks. *IEEE Wirel. Commun.*, 12: 69-81.
- Jiang, M., J. Li and Y. Tay, 1998. Cluster based routing protocol (CBRP) functional specification. IETF Internet-Draft. <http://www3.ietf.org/proceedings/98dec/slides/manet-cbrp-98dec/>.
- KaiTen, F., H. Chung Hsien and L. TseEn, 2008. Velocity-Assisted Predictive Mobility and Location-Aware Routing Protocols for Mobile Ad Hoc Networks. *IEEE Trans. VEH. Technol.*, 57: 448-464.
- Karp, B. and H.T. Kung, 2000. GPSR: Greedy Perimeter Stateless Routing for Wireless Networks. *Proceedings of MobiCom 2000, 6th Annual International Conference on Mobile Computing and Networking*, 2000 Boston, MA, pp: 243-254..
- Ko, Y.B. and N.H. Vaidya, 2000. Location-aided routing (LAR) in mobile ad hoc networks. *Wirel. Network*, 6: 66-75.
- Kuruwila, J., A. Nayak and I. Stojmenovic, 2005. Hop count optimal position-based packet routing algorithms for ad hoc wireless networks with a realistic physical Layer. *IEEE J. Sel. Area Commun.*, 23: 1267-1275.
- Ritchie, L., H. Yang, A. Richa and M. Reisslein, 2006. Cluster Overlay Broadcast (COB): MANET Routing with Complexity Polynomial in Source-Destination Distance. *IEEE Trans. Mobile Comput.*, 5: 653-666.
- Theoleyre, F. and F. Valois, 2005. Virtual Structure Routing in Ad hoc Networks. *Proceedings of IEEE International Conference on Communications 2005*, 2005 Seoul, Korea, pp: 3078-3082..
- Wang, L. and S. Olariu, 2004. A Two-Zone Hybrid Routing Protocol for Mobile Ad Hoc Networks. *IEEE Trans. Parall. Distr.*, 15: 1105-1116.
- Woo, S.C.M. and S. Singh, 2001. Scalable Routing Protocol for Ad Hoc Networks. *Wirel. Network*, 7: 512-529.