

Stability Aware Cluster Routing Protocol for mobile Ad-Hoc Networks

Chun-Yuah Chiu, Eric Hsiao-kuang Wu, and Gen-Huey Chen

Department of Computer Science and Information Engineering

National Taiwan University, Taipei, Taiwan

f88056@csie.ntu.edu.tw, hsiao@csie.ncu.edu.tw, ghchen@csie.ntu.edu.tw

Abstract- An ad-hoc network is formed by a collection of mobile nodes without any centralized access point or existing infrastructure. Communication between mobile nodes requires routing over the multiple-hop wireless path. Since the mobile nodes could be of high mobility, the effective and adaptive routing protocol must have on-going detail of the topology information. However, it wastes limited bandwidth to keep routing information up-to-date and reliable. Thus, a crucial algorithm design objective to achieve routing responsiveness and updating efficiency is the minimization of reaction to mobility.

This paper proposes an efficiently repairable routing protocol, called as gravitational cluster routing (GCR) protocol. It is based on a stable cluster structure that covers dense areas to increase stability, and avoids articulate nodes in the cluster to connect strongly and adopts unicast to minimize reaction to mobility. The active routing paths of GCR can be maintained locally by individual stable clusters and globally by quantifying the corresponding repairable levels to satisfy the distinct demand parameters of QoS. These mechanisms improve the stability of active connections.

1. INTRODUCTION

The existing wireless networks can be classified into two main types, infrastructure and infrastructure-less mobile networks. In recent years, the rapid growth of wireless communication has offered various services and conveniences to the people. Many of the services are delivered through fixed network as their backbone. However, there are several applications such as disaster rescues, tactical communications for military usages, and wireless conferences for large-scale meetings, where the fixed communication infrastructure is hard to establish. So the ad-hoc network is evolved to accommodate the situation. An ad-hoc network is formed by a collection of mobile nodes without any centralized access point or existing infrastructure. Communication between mobile nodes requires routing over the multiple-hop wireless path. Distinct from traditional wired networks or infrastructure networks, frequent link failures in wireless environment and infrastructure-less characteristics bring the new challenges to design an effective and adaptive routing protocol. Many restrictions should be well considered such as high power consumption, limited bandwidth to reduce the feasibility of routing protocol. Furthermore, the rapid advances in

hardware design have greatly reduced cost, size and the power requirements of network elements. The large and dense wireless mobile networks such as wireless sensor networks are expected to find wide applicability. Therefore, a scalable and QoS guaranteed routing protocol is necessary for multimedia applications in such wireless networks.

Many routing protocols have been developed for ad hoc networks. The evolution of routing protocols can be categorized to three phases:

- In the first phase, a wireless-oriented routing protocol is desirable since the traditional routing protocols for wire networks such as distance Bellman-ford [1] or link state [2] are not suitable for wireless environment. Many table-driven and on-demand routing protocols have been proposed, such as Destination-Sequenced Distance Vector (DSDV) routing protocol [3], Temporally-Ordered Routing Algorithm (TORA) [4], Dynamic Source Routing (DSR) protocol [5], and Ad-hoc On-Demand Distance Vector (AODV) routing protocol [6], etc.
- In the second phase, an efficient and responsive route is expected since the table-driven routing protocols waste limited bandwidth to continuously keep routing information up-to-date and the on-demand routing protocols need longer route acquisition latency. The hierarchical and hybrid routing protocols have been proposed, such as Zone Routing Protocol (ZRP) [7], Dynamic Group Routing [8], Distributed Dynamic Routing (DDR) Algorithm [9], Access-Based Clustering Protocol (ABCP) [10] and [11-13], etc.
- In the third phase, the route acquisition is not the only issue in ad-hoc network. In order to satisfy the special application and achieve the requirement of QoS, the high power consumption, limited bandwidth, asymmetric link, and a variety of QoS will be considered in the routing protocol. There are many novel routing protocols have been proposed, such as Multiple Next Hops (MNH) routing protocol [14], Ad hoc on-demand Backup node setup Routing Protocol (ABRP) [15], and [16], etc.

The end-to-end perceived quality of the application is closely relative to the stability of the active connection. Many previous protocols focus on reconnection as link failure occurs. However, they are not suitable for large ad hoc networks since the number of hops in the expected route is of order \sqrt{n} . The hierarchical and hybrid structures can

satisfy the scaling problem [17], since the cluster-based routing can make a large network appear smaller and a highly dynamic topology appear much less dynamic [11]. However, the influence on the wireless links due to the mobility of ad hoc networks is not reduced. The mobility problem is just transformed to the stability problem of the cluster. The repair cost of routing path will be enforced by the maintenance cost of hierarchical structure. The previous hierarchical routing protocols change their hierarchical structures to react any topology change by locally broadcasting even though some topology changes might not alter the hierarchical structures. The mobility of nodes causes the change of link states in their neighborhood. We can regard the mobility of clusters as the change of link states in cluster level. Therefore, the stability and mobility of clusters will be trade-off in the cluster scheme. This is because that the density of nodes in the ad hoc networks may not be uniform all the time. If the probability of each link failure is assumed to be equal, clusters that cover the dense area will be more stable and of higher mobility (the number of links between clusters is sparse). On the other hand, clusters that cover the sparse area are more instable with lower mobility.

The proposed Gravitational Cluster Routing (GCR) protocol is based on a hierarchical clustering structure. It has been designed specially to improve the end-to-end perceived quality of the application by increasing the stability of active connection. The cluster scheme is to cover dense areas to increase stability, and avoid articulate nodes in the cluster to connect strongly and use unicast to minimize reaction to mobility. This policy may result in higher mobility of cluster. So GCR quantifies the repairable level of a route (maximum flow [19]) and select different repairable levels of routes to satisfy the different demand of end-to-end perceived quality of the application specifically. In the simulation study, we demonstrate that GCR is an efficient and inexpensive routing protocol and is more suitable for large and dense ad hoc networks than zone routing protocol. The proposed GCR protocol carries the following advantages:

- Providing end-to-end perceived QoS guarantee
- Constructing a QoS aware cluster backbone to support the upper layer protocols
- suitable for large and dense ad hoc networks

The remainder of this paper is organized as follows. Section 2 describes some related assumption of link protocol. Section 3 describes the GCR in detail. Section 4 provides some numerical analysis and simulation results. The paper then concludes with challenges and future works.

2. SYSTEM MODEL

In an ad-hoc network, each mobile node has a unique node identifier, denoted as *ID*, and communicates with each other by radio. We assume that the radio transmitting range

of each node is equal, and define that the neighbors of a node *u* are a set of nodes being covered by the transmitting range of *u*, denoted as *N(u)*. Therefore, in this system model there exist only the symmetrical links between nodes and their neighbors. We assume each node maintains a “Link Information Table” to record the state of its neighbors, and will broadcast a control signal named as “Beacon” periodically to update Link Information Tables. Furthermore, since the proposed GCR is the member of the clustering routing protocols, each node will be required to maintain some data for cluster maintenance. Definitions are as follows: *deg(u)* is the number of node *u*’s neighbors, *Cid(u)* is the *ID* of the cluster head that dominates node *u*, *STid(u)* is the *ID* of a node which is the neighbor of the cluster head and lies on the shortest path, *p(u)* (*p(u)* is defined in Section 3.3), and *l(u)* is the hop count from node *u* to its cluster head.

NodeId	ClusterId	level	Stable
--------	-----------	-------	--------

The fields of Beacon

NodeId	ClusterId	level	stable	SubtreeId	Degree
--------	-----------	-------	--------	-----------	--------

The fields of Link Information Table

3. ROUTING PROTOCOL

3.1 Main Ideas

The goal of this approach is to construct a quickly repairable routing path with light cost in maintenance. In order to achieve this goal, we divide the network into several non-overlapping clusters and adopt proactive routing protocol inside the clusters and reactive routing protocol outside the clusters. When two mobile hosts communicate to each other, the routing path connecting these two mobile nodes may be through several intermediate clusters. Therefore, the maintenance efforts of the entire routing path can be distributed into several intermediate clusters.

GCR avoids articulate nodes in the cluster on the cluster construction phase to reduce clustering maintenance cost and connect strongly [20]. Nodes that have “local maximum degree” are selected to cover the dense area forming stable clusters. Besides, the cluster structure is designed as a gravitational area to reduce broadcasting control packets for cluster maintenance. Each node of clusters knows its potential of the gravitational area, and only the cluster head knows the entire cluster topology. The active connection can be repair efficiently and locally by light maintenance cost and stable clusters. In order to achieve global repair efficiently, GCR quantifies the repairable level of a route and use it to find routes that have stronger connectivity between clusters to overcome the high mobility of clusters. The relation between the above strategies and their advantages are in Fig. 1.

3.2 Cluster Head Selection

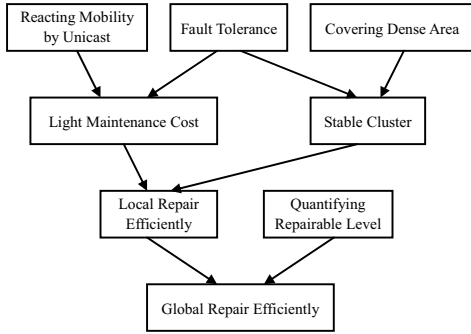


Fig.1 The relation between strategies of GCR and their advantages

When the cluster id of a node and all its neighboring nodes are null over a particular periodic time, the node broadcasts degree information to its all neighbors. The node that is not a member of any cluster and receives the degree information from its neighbor will broadcast its degree information to its all neighbors again (see Fig.3). After this procedure, the node whose degree is greater than its all neighbors will be a cluster head. The detailed procedure is as follows:

```

If ( Cid(u) is NULL )
  If (  $\forall$  node  $v, v \in N(u), Cid(v)$  is NULL or  $l(v)$  is equal to
    MAXIMUM CLUSTER RADIUS )
    Node  $u$  broadcasts  $deg(u)$  to all nodes of  $N(u)$ ;
  
```

Furthermore, we can consider the stability of nodes [18] in the cluster head election. It can reduce the probability of the cluster head failure to avoid the high cost of cluster reconstruction.

3.3 Cluster Construction

The Cluster Construction Procedure (see Fig.2) can be described in two parts. In the first part, after the cluster head election, the cluster head broadcasts “gravitation packet” locally (in MAXIMUM CLUSTER RADIUS). When a node u receives the “gravitation packet”, it will know its potential of the gravitational area named as *level* (the hop count from cluster head to u) and *SubtreeId* (see Fig.4). The “gravitation packet” which decides $l(u)$ is forwarded through a shortest path from the cluster head to node u . Such shortest path could be denoted as $p(u)$. The *SubtreeId* is the ID of a node which is the neighbor of cluster head and lies on the shortest path, $p(u)$. With the help of *SubtreeId*, the GCR can avoid articulate nodes in the cluster on cluster construction phase.

<i>ClusterId</i>	<i>level</i>	<i>SubtreeId</i>
------------------	--------------	------------------

The fields of Gravitation packet

In the second part, each node that receives the “gravitation packet” decides if it is “jointless” or not. If the *SubtreeId* of $N(u)$ which have same *ClusterId* with node u

are all same, the node u is not “jointless”. Else the node u is “jointless”. A node is *stable* if it is dominated by a cluster head and has one up-level neighbor at least. In the cluster construction phase, if a node u is unstable and does not have any neighbor which is stable and its level is $l(u)$ plus one, the node u is the leaf node. If a leaf node u is jointless, it will send the “join packet” which includes the link information of node u through path $p(u)$ to the cluster head. Else, the leaf node will give up joining the cluster. When each node receives the “join packet”, it will append its link information to the “join packet” and forward “join packet” to the up-level neighboring nodes until the cluster head receives the packet. So, the cluster head knows the entire cluster topology and each node knows its level of the cluster after the cluster construction is performed. We will prove that such clusters can tolerate one node fault unless the faulty node is cluster head in Theorem 1.

```

Update SubtreeId field in the node  $v$ 's entry of Link Information Table;
if (  $l(u) = 0$  and  $Cid(u) = \text{NULL}$  )
  level++;  $l(u) = \text{level}$ ;  $Cid(u) = \text{ClusterId}$ ;
  if ( level = 1 )
    STid(u) = Id(u);
    SubtreeId = Id(u);
  else STid(u) = SubtreeId;
  if ( level < MAXIMUM CLUSTER RADIUS )
    broadcast “gravitation packet”;
  else discard “gravitation packet”;
  
```

Fig.2-1 When a node u receives “gravitation packet” from node v

```

if ( the SubtreeId of  $N(u)$  which have same ClusterId with node  $u$  are
  all equal )
  the node  $u$  is not “jointless”;
else the node is “jointless”;
  
```

Fig.2-2 Each node decides if it is “jointless” or not

```

if ( the node is leaf node )
  if ( the node is “jointless” )
    reply “join packet” to up-level neighbor;
    // | NodeId | Link Information |
    set it to “stable”;
  else  $l(u) = 0, Cid(u) = \text{NULL}, STid(u) = \text{NULL}$ ;
    // give up joining the cluster
  
```

Fig.2-3 Leaf node replies “join packet” to cluster head

```

if (  $l(u) \neq 0$  ) // not cluster head receive
  if (  $u$  is not “stable” )
    append its link information to “join packet”;
    set  $u$  to “stable”;
    reply “join packet” to up-level neighbor;
  else record the total link information of the “join packet”;
    // cluster head receive
  
```

Fig.2-4 When a node u receives “join packet”

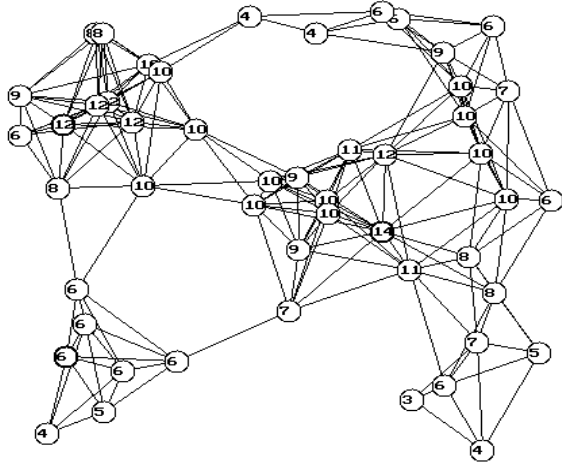


Fig.3 An ad-hoc network consists of 50 nodes. The number of the node is the degree of itself and the nodes with thick lines have the local maximum degree.

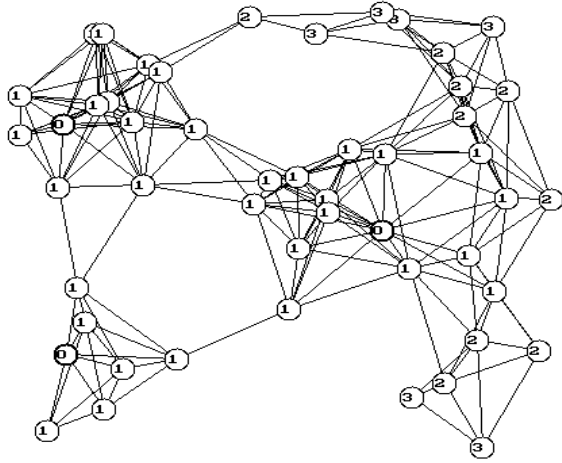


Fig.4 Each node receives the "gravitation packet" and knows its level

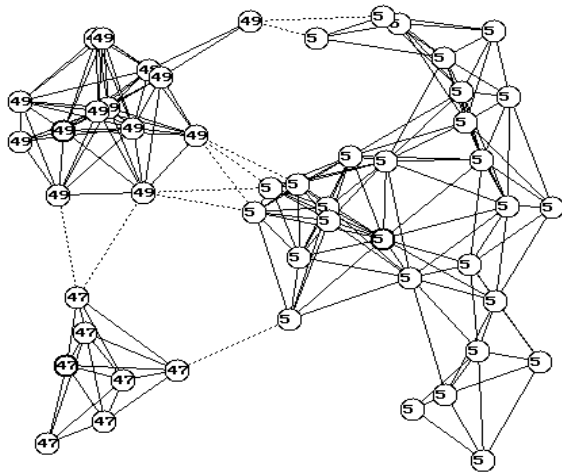


Fig.5 Clusters have been constructed in GCR. The numbers are the Cluster ID of nodes

THEOREM 1. Through the above Cluster Construction Procedure, the cluster can tolerate one node fault unless the faulty node is cluster head.

Proof. We assume that cluster C is established by Cluster Construction Procedure and tree $T(C)$ which includes total nodes in C and roots at cluster head is formed by all $p(u)$. For each node in $T(C)$, when a particular node u (not cluster head) is failure:

Case 1: If u is a leaf node, then $C - \{u\}$ is connected. #

Case 2: If u is not a leaf node, we assume that w is the parent of u and $v_1 \sim v_k$ are k children of u . We only need to prove that there exist paths from w to $v_1 \sim v_k$:

If v_1 is "jointless", there is at least an existing neighbor u' and $STid(v_1)$ is different from $STid(u')$. So we can construct a path from w to v_1 : $p(w) + p(u') + (u', v_1)$. (see Fig.6(a)) // $p(u')$ is existed, because u' is also "jointless" and u does not lie on $p(u')$

Else v_1 must be not a leaf node, and there is at least an existing descendant leaf node v' is "jointless". Since v' is "jointless", there is at least existing a neighbor u' and $STid(v')$ is different from $STid(u')$. So we can construct a path from w to v_1 : $p(w) + p(u') + (u', v') + (p(v') - p(v_1))$. (see Fig.6(b)) // $p(u')$ is existed, because u' is also "jointless" and u does not lie on $p(u')$.

$v_2 \sim v_k$ is similar. #

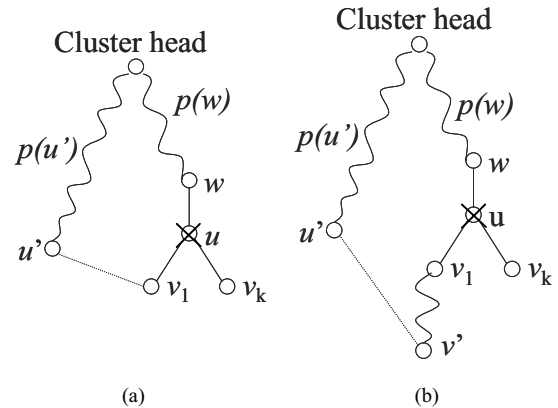


Fig.6 The illustration of theorem 1

3.4 Cluster Maintenance

In an ad-hoc network, the network topology is changed frequently by some events among mobile nodes, such as the mobility and switching. In order to reduce the cost of cluster maintenance, the frequent change of cluster structures to react the change of network topology is not necessary. So the GCR does not maintain the fault tolerance feature of clusters expressly in cluster maintenance. It only guarantees that each node of the network is stable. On the other hand, unlike the previous clustering routing protocol, in GCR only the cluster head maintains the topology information of the

cluster and other nodes do not maintain the duplicate topology information. Therefore, GCR can utilize unicasting instead of broadcasting to react the topology change. The “Cluster Maintenance Procedure” (see Appendix) can be described in two situations.

Link or node failure:

When the link (u,v) is failure, the upper node (either u or v) which has smaller level will send a “link change packet” to notify the cluster head. If the lower one has other stable neighbors, it will select a stable neighbor k which has minimum level and change its level to $l(k)$ plus one. Otherwise, the lower node will separate itself from the cluster and wait a chance to rejoin a cluster or reinitiate to construct a new cluster. Since the link failure may change the level of the lower node, this change may propagate until all nodes of the cluster are stable. We name such link as the *influential link*. It may cause some nodes to separate themselves from the cluster and the ad hoc network will pay high cost for maintenance. We will prove the ratio of influential links to total links is at most “ $2/Average Degree$ ” in Theorem 2. The node failure can be regarded as several link failures. In this case, we know the cluster head failure will result in heavy maintenance cost to reconstruct the cluster. Therefore, the proposed cluster head election algorithm can evaluate the stability of nodes [18] to reduce the probability of the cluster head failure.

Link connected or node switched on:

When the node u and node v are connected, the node u which has smaller level will send a “link change packet” to notify the cluster head. The node v will change $l(v)$ to $l(u)$ plus one if $l(v)$ is greater than $l(u)$ plus one. Similarly this change may propagate until all nodes of cluster are stable. When a new node switches on, it will wait a chance to join a most suitable cluster or initiate to construct a new cluster. The most suitable cluster can be obtained according to the level that will be obtained of this joining node and the connectivity between this node and this cluster.

THEOREM 2. The ratio of influential links to total links is at most “ $2/Average Degree$ ”.

Proof. A node u updates its $l(u)$ only if it has no neighbors in $l(u)-1$ level. Therefore, we consider that tree $T(C)$ is a spanning tree of cluster C defined as before. Links of $T(C)$ be named as tree links and others of C be named as non-tree links. If a non-tree link (u,v) is failure and $l(u)$ is great than $l(v)$, node u would not update its $l(u)$ because there is existing a tree link (u,v') and $l(v')$ is $l(u)-1$. Thus only tree link failure may cause many nodes to change their levels. Some inequalities can be derived as follow:

$$\text{“The number of influential links”} < \text{“The number of tree links”} \quad \dots a$$

$$\begin{aligned} \text{“The number of tree links”} &= \text{“The number of nodes”} - \\ \text{“The number of clusters”} &\quad \dots b \end{aligned}$$

$$\begin{aligned} \text{“The number of total links”} &= \text{“The number of nodes”} \times \\ \text{“Average Degree / 2”} &\quad \dots c \end{aligned}$$

From b, c

$$\begin{aligned} \text{The ratio of tree links to total links (denoted as } r_T) &< \\ \text{“2/Average Degree”} &\quad \dots d \end{aligned}$$

From a, d

$$\begin{aligned} \text{The ratio of influential links to total links (denoted as } r_I) &< \\ \text{“2/Average Degree”} &\quad \# \end{aligned}$$

3.5 Routing

After the cluster construction in section 3.3, the network is partitioned into a set of non-overlapping clusters. The cluster head maintains the topology information of its cluster and the link information between itself and its neighboring clusters. The other nodes do not maintain any routing table. When a node tends to communicate with other nodes in this ad hoc network, it will send route request packet (RREQ) to the up-level neighboring node and the RREQ is forwarded to the cluster head step by step. When the cluster head receives RREQ, it will broadcast RREQ in super node level. The inter-cluster communication can be described as follows. First a cluster head transmits packets to its bordering nodes by unicast. Second the particular bordering node of another cluster receives the packets and then forwards them to its cluster head according to the level of each node step by step. After the cluster head that dominates the destination node receiving the RREQ, it will reply route reply packet (RREP) to the source node and the route from source to destination is constructed.

When two mobile hosts communicate to each other, we denote the passing clusters from source to destination as C_1, C_2, \dots, C_i , and the two bordering nodes of C_i which are connected with C_{i-1} (upstream cluster) and C_{i+1} (downstream cluster) as U_{C_i} and D_{C_i} . If the communication is disconnected due to the mobility of the ad hoc network, the “Route Maintenance Procedure” can be described in two cases:

Case 1. The disconnection is due to the inter-cluster link between C_i and C_{i+1} failure. If there are other inter-cluster links between same clusters, the cluster head of C_i will reconstruct the path through them. Else, each cluster can be regarded as a super node and the strategy of reconstruction is similar to AODV.

Case 2. The disconnection is due to the intra-cluster link of C_i failure. According to the description of section 3.4, the Cluster Maintenance Procedure will be executed. After that, if the C_{i-1} and C_{i+1} are also neighbors of this cluster, the path is maintained automatically. Else, the strategy of reconstruction is similar to AODV in super node level.

The cluster structure of GCR has high stability and low maintenance cost obviously as previous description. It can be a QoS aware backbone for routing. The routing path that is formed by above may not be a shortest path, so GCR can reduce it with partial optimization inside the clusters. Because the cluster head maintains the network topology of its cluster, GCR can optimize the intra-cluster paths from Uc_i to Dc_i by any known shortest path algorithm. Furthermore, in order to provide end-to-end perceived QoS guarantee, GCR quantifies the repairable level of a routing path to satisfy the different demand parameters of QoS. The cluster C_i can be seen a graph and the repairable level of a routing path P is the connectivity from source to destination in the graph $C_1+C_2...+C_t$, denoted as $RL(P)$. In another word, $RL(P)$ is a number of links in minimum cut set [19] from source to destination in graph $C_1+C_2...+C_t$. We denote:

$L(C_i, C_{i+1})$ is a set of inter-cluster links connected between C_i and C_{i+1}

Bc_i, C_{i+1} is a set of bordering nodes of C_i connected to C_{i+1}

$MCS(s, t)$ is a number of links in minimum cut set from s to t , s and t are sets of nodes

Then:

$$RL(P) = \text{MIN} \{ \text{MIN} \{ \#L(C_i, C_{i+1}), i = 1 \sim t-1 \}, MCS(\text{source}, Bc_1, C_2), \text{MIN} \{ MCS(Bc_i, C_{i-1}, Bc_i, C_{i+1}), i = 2 \sim t-1 \}, MCS(Bc_t, C_{t-1}, \text{destination}) \}$$

So GCR can obtain a routing path to satisfy the different demand parameters of QoS by using $RL(P)$ in route discovery phase.

4. NUMERICAL ANALYSIS AND SIMULATION RESULT

In Theorem 2, we know that the ratio of influential links to total links is at most “ $2/\text{Average Degree}$ ”. In an ad-hoc network environment, we assume that the transmitting range of each node is a circular area. The degree of a node can be considered that how much transmitting ranges of other nodes cover the position of this node. So the average degree of total nodes in an ad-hoc network can be considered the sum of areas that the transmitting range of total nodes covered divided by the area of entire ad-hoc network environment. We assume that the ad-hoc network environment is a rectangular area and a node would be placed randomly in this rectangular area with equal probability. So the average degree is approximate:

$$\left(lw - \frac{r(l+w)}{2} + r^2 \right) \times \frac{n\pi r^2}{(lw)^2}$$

l : the length of ad-hoc network environment
 w : the width of ad-hoc network environment
 r : the radius of transmitting range of each node
 n : the number of nodes

According to this result, we can calculate the ratio of influential links to total links before simulation or implementation. Fig.7 shows the analytic r_T , actual r_T , and actual r_l (define in theorem 2) with different transmitting radius. We can know that the analytic r_T is exact and Theorem 2 is correct. Similarly, Fig.8 show same value with different number of total nodes.

According to Fig.7 and Fig.8, the r_l is decreasing when the average degree is increasing. In another word, the GCR is more suitable for large and dense ad hoc networks. Fig.9 shows the routing path length of pure GCR, GCR with local optimization, and the shortest path with different cluster hop count (the number of clusters that the routing path passes through). The ratio of routing path length of pure GCR to the shortest path approximates 1.4. After performing local optimization, the ratio can be improved to approximate 1.1. Fig.10 shows the successful rate of different QoS requirement with different average degree. We define that the inter-cluster link connectivity of C_i and C_j is the number of links in $L(C_i, C_j)$. The average inter-cluster link connectivity is the number of total inter-cluster links divided by the number of total super links (regarded clusters as super nodes). In Fig.10, we can know that the successful rate of the QoS requirement i is increasing quickly after the average inter-cluster link connectivity over i . Because the connectivity inside the clusters is stronger than the inter-cluster link connectivity, the inter-cluster link connectivity becomes the main factor in $RL(P)$.

In Fig.11 and Fig.12, GCR has been compared with the typical hierarchical routing scheme, zone routing protocol (ZONE RADIUS is 3). Fig.11 shows the average number of maintenance packets of GCR and zone routing protocol respectively with different average degree. It is obvious that the maintenance cost of zone routing protocol is increasing as the density of ad hoc network is increasing. This is because zone routing protocol broadcasts maintenance packets locally to react any topology change. Relatively GCR has more efficient behavior in dense ad hoc networks. We define that the repair time of a faulty routing path in a transmission pair is the additional delay time when the destination receive the data packets. Fig.12 shows the average repair time of GCR and zone routing protocol respectively with different average degree. The repair time of GCR and zone routing protocol are decreasing as the density of ad hoc networks is increasing. It is because that relaying nodes of routing paths can always bypass link failures in dense ad hoc networks.

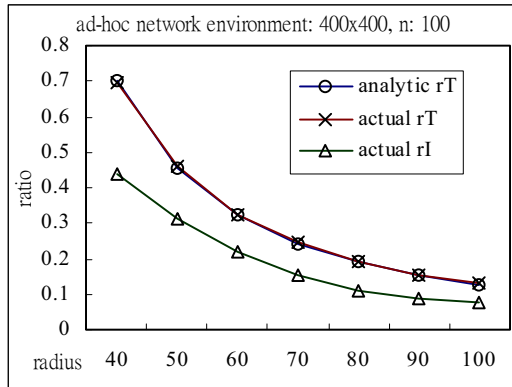


Fig.7 Analytic r_T , actual r_T , and actual r_I with different transmitting radius

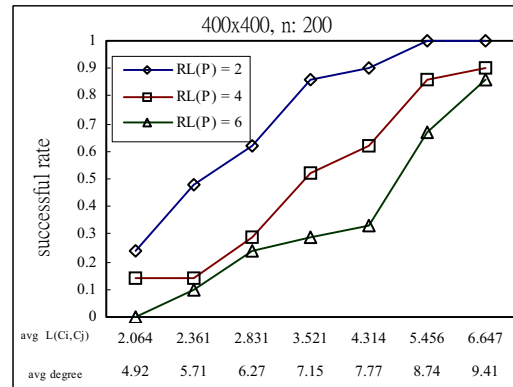


Fig.10 Successful rate of different QoS requirement with different *Average Degree* and average inter-cluster link connectivity

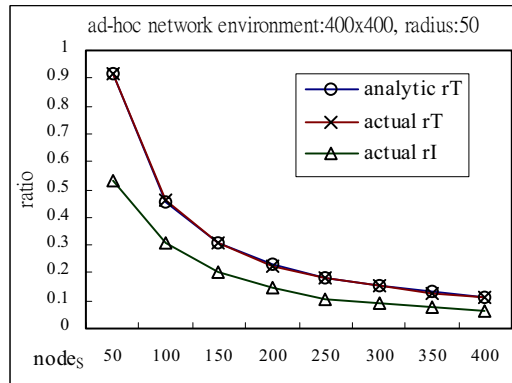


Fig. 8 Analytic r_T , actual r_T , and actual r_I with different number of total nodes

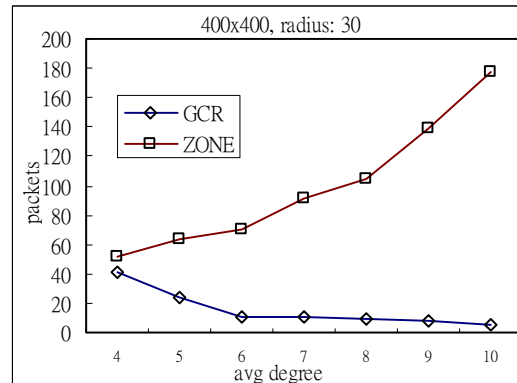


Fig.11 The average number of maintenance packets of GCR and zone routing protocol with different average degree

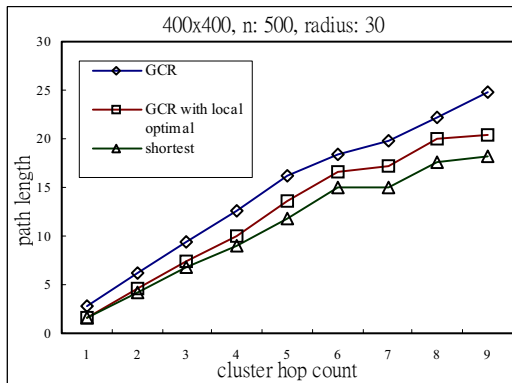


Fig. 9 The routing path length of pure GCR, GCR with local optimization, and the shortest path with different cluster hop count

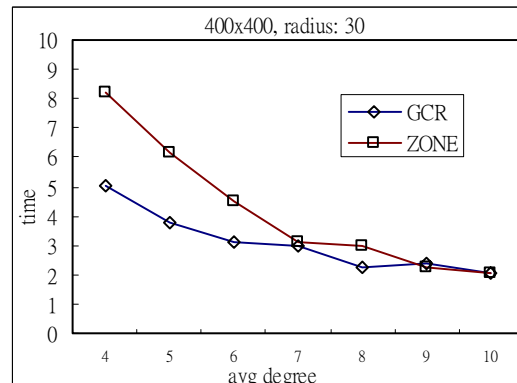


Fig.12 The average repair time of GCR and zone routing protocol with different average degree

5. CONCLUSION

In this paper, we have proposed an efficiently repairable routing protocol, called as gravitational cluster routing (GCR) protocol. The major differences between GCR and the previous clustering routing protocol are that the cluster structure in GCR has two special features, fault tolerance and no duplicate topology information. Therefore such cluster structure can be maintained efficiently and be a QoS aware backbone for the upper layer protocols. On the other hand, in order to enhance the end-to-end perceived QoS, we exploit the above QoS aware backbone to reduce the routing path length and quantify the repairable levels of routing paths to satisfy the higher requirement of QoS. Finally, in order to demonstrate the feasibility of our work, we propose two theorems and some simulation results to prove that the cluster structures of GCR are stable and inexpensive. Furthermore, we compare the repair cost and repair time of GCR and routing zone based route maintenance techniques by simulation. GCR is more suitable for large and dense ad hoc networks than zone routing protocol by the simulation results. For future work, we will consider the stability of nodes [18] or more factors in the cluster head election to construct stronger (stable) cluster structures. More different demands of QoS will be considered to embed in the cluster backbone. Furthermore, our work could be integrated with other heterogeneous networks.

ACKNOWLEDGEMENT

This research was supported by the Communications Software Technology project of Institute for Information Industry and sponsored by MOEA, R.O.C

REFERENCES

- [1] D.Bertsekas and R.Gallager, *Routing in data networks*, chapter5, Prentice Hall, second edition, 1987.
- [2] J.M.McQuillan et al., "The new routing algorithm for the ARPANET," *IEEE Transaction of Communications*, pp.711-719, May 1980
- [3] C.Perkins and P.Bhagwat, "Highly dynamic destination-sequenced distance vector routing (DSDV) for mobile computers," *ACM SIGCOMM*, October 1994
- [4] V.D.Park and M.S.Corson, "A highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks," *IEEE INFOCOM*, April 1997
- [5] D.B.Johnson and D.A.Maltz, "Dynamic Source Routing in Ad-Hoc Wireless Networks," *Mobile Computing*, edited by T.Imielinski and H.korth, Kluwer Academic Publishers, 1996
- [6] C.E.Perkins and E.M.Royer, "Ad-hoc On-Demand Distance Vector Routing," *IEEE WMCSA*, 1999
- [7] M.R.Pearlman and Z.J.Hass, "Determining the Optimal Configuration of the Zone Routing protocol," *IEEE JSAC*, VOL. 17, NO. 6, August 1999
- [8] Y.L.Chang and C.C.Hsu, "Routing in wireless/mobile ad-hoc networks via dynamic group construction," *ACM Mobile Networks and Applications*, 2000
- [9] N.Nikaein, H.Labiod and C.Bonnet, "DDR-Distributed Dynamic Routing Algorithm for Mobile Ad hoc Networks," *IEEE MobiHOC*, 2000
- [10] Ting-Chao Hou and Tzu-Jane Tsai, "An Access-Based Clustering Protocol for Multihop Wireless ad hoc Networks," *IEEE JSAC*, VOL. 19, NO.7, July 2001
- [11] A.B.McDonald, T.F.Znati, "A Mobility-Based Framework for Adaptive Clustering in Wireless Ad Hoc Networks," *IEEE JSAC*, VOL. 17, NO. 8, August 1999
- [12] D.Kim, S.Ha, and Y.Choi, "K-hop Cluster-based Dynamic Source Routing in Wireless Ad-Hoc Packet Radio Network," *IEEE VTC*, 1998
- [13] Suman Banerjee and Samir Khuller, "A Clustering Scheme for Hierarchical Control in Multi-hop Wireless Networks," *IEEE INFOCOM*, 2001
- [14] M.H.Jiang and R.H.Jan, "An Efficient Multiple Paths Routing Protocol for Ad-hoc Networks," *IEEE Information Networking*, 2001
- [15] C.M.chung, Y.H.Wang, and C.C.Chuang, "Ad Hoc On-Demand Backup Node Setup Routing Protocol," *IEEE Information Networking*, 2001
- [16] J.Wu and H.Li, "Domination and Its Applications in Ad Hoc Wireless Networks with Unidirectional links," *IEEE Parallel Processing*, 2000
- [17] Marc R. Pearlman, Zygmunt J. Hass and Syed I. Mir, "Using Routing Zones to Support Route Maintenance in Ad Hoc Networks," *IEEE WCNC*, 2000
- [18] C.K.Toh, "A Novel Distributed Routing Protocol To Support Ad-Hoc Mobile Computing," *IEEE Computers and Communications*, 1996.
- [19] T.H.Cormen, C.E.Leiserson and R.L.Rivest, *Introduction to algorithm*, chapter27, McGraw-Hill Book Company, twentieth printing, 1998
- [20] F.Buckley and F.Harary, *Distance in Graphs*, Addison-Wesley, MA, 1990