

PAPER

Multipath Routing with Reliable Nodes in Large-Scale Mobile Ad-Hoc Networks

Yun GE[†], Guojun WANG^{†*a)}, Qing ZHANG^{††}, *Nonmembers*, and Minyi GUO^{†††}, *Member*

SUMMARY We propose a Multiple Zones-based (M-Zone) routing protocol to discover node-disjoint multipath routing efficiently and effectively in large-scale MANETs. Compared with single path routing, multipath routing can improve robustness, load balancing and throughput of a network. However, it is very difficult to achieve node-disjoint multipath routing in large-scale MANETs. To ensure finding node-disjoint multiple paths, the M-Zone protocol divides the region between a source and a destination into multiple zones based on geographical location and each path is mapped to a distinct zone. Performance analysis shows that M-Zone has good stability, and the control complexity and storage complexity of M-Zone are lower than those of the well-known AODVM protocol. Simulation studies show that the average end-to-end delay of M-Zone is lower than that of AODVM and the routing overhead of M-Zone is less than that of AODVM.

key words: mobile ad-hoc networks, multipath routing, node-disjoint paths, multiple zones, reliable nodes

1. Introduction

A mobile ad-hoc network (MANET) consists of a set of mobile nodes that can communicate with each other using multi-hop wireless links. Due to node mobility, the network topology changes dynamically and traditional routing protocols cannot be used directly in MANETs.

Existing routing protocols are generally classified into proactive routing [1], reactive routing [2], [3], and hybrid routing [4]. As the development of global positioning system (GPS) technique, location-based routing protocols are presented to forward packets according to geographic location information [5]–[8].

However, the routing path in the above single path routing is easy to break and thus result in frequent route rediscovery. Recently many multipath routing protocols have been proposed, which can be categorized into non-disjoint multipath routing, link-disjoint multipath routing, and node-disjoint multipath routing [9]. Non-disjoint paths can have nodes and links in common. Link-disjoint paths do not have

any links but have some nodes in common. Node-disjoint paths do not have any node in common except for the source and the destination, so the failure of a node will not affect other paths except for the path where the node resides. Node-disjoint multipath routing offers the most aggregate resources and the highest fault-tolerance, but it is hard to find them.

Most multipath routing protocols are based on single path routing protocols. Split multipath routing (SMR) [10] and multipath source routing (MSR) [11] are extensions to DSR. SMR finds an alternate route that is maximally disjoint from the shortest delay route between the source and the destination. MSR distributes load among multiple paths based on the measurement of round trip time (RTT). Dong and Puri [12] present a multipath routing protocol based on DSDV. Ad-hoc on-demand multipath distance vector (AOMDV) [13] and ad-hoc on-demand distance vector multipath (AODVM) [14] are extensions to AODV. In AOMDV, intermediate nodes do not discard duplicate copies of a route request (RREQ) packet immediately and only link-disjoint paths are guaranteed. The aforementioned multipath routing protocols are the typical protocols used to find disjoint paths, but none of them can guarantee that node-disjoint paths will be found except for AODVM.

Summarizing the above techniques, none of them makes full use of geographic location information to discover node-disjoint multipath routing effectively. Location-based routing protocol has good scalability, thus we propose a Multiple Zones-based routing protocol (M-Zone) to discover node-disjoint multiple paths in large-scale MANETs using a location-based multiple zoning method.

A preliminary version of this paper [17] has been published in HPCC 2008, and the approach used there works well when the nodes are distributed uniformly and randomly with comparatively high node density. But it is not suitable for the networks with holes (e.g., the sparse areas). This paper deals with more general network environments borrowing the idea of placing reliable nodes from AODVM.

The remainder of this paper is organized as follows. Section 2 introduces the reliable nodes deployment in AODVM. Section 3 describes the network model. Section 4 presents the design of the proposed M-Zone protocol. Performance analysis is presented in Sect. 5 and simulation studies are presented in Sect. 6. Finally, we conclude the paper in Sect. 7.

Manuscript received December 4, 2008.

Manuscript revised April 12, 2009.

[†]The authors are with the School of Information Science and Engineering, Central South University, Changsha 410083, Hunan Province, China.

^{††}The author is with the School of Information Engineering, Dalian Maritime University, Dalian 116026, Liaoning Province, China.

^{†††}The author is with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200030, China.

*The corresponding author.

a) E-mail: csjgwang@mail.csu.edu.cn

DOI: 10.1587/transinf.E92.D.1675

2. The Min-Cut Based R-Node Deployment Strategy in AODVM

AODVM is a well-known node-disjoint multipath routing protocol in MANETs. As the distance between a source and a destination increases, bottlenecks (for example the sparse areas) inevitably occur and thus, it is very hard to discover node-disjoint multiple paths. Reliable nodes (R-nodes for short) are placed to support finding multipath routing in AODVM. R-nodes are highly reliable, powerful and secure nodes located in a tank or any other large vehicles. These nodes move faster than normal nodes, however, it would be too expensive to deploy too many R-nodes in the network.

How to identify positions for the R-nodes such that the routing paths are more reliable? AODVM uses a modification of the randomized min-cut algorithm [16] to handle this problem. AODVM assumes that each node knows its own coordinates by using GPS or other techniques. Each node knows its k -hop local topology (k is a system parameter). Then each node computes its min-cut value of its k -hop local topology according to the contraction algorithm with the following modification: the outermost links are contracted first, and the links that are closest to the node are contracted last. The modification is to ensure that the min-cut value is an accurate indicator of the importance of the computing node in keeping the local topology connected. The smaller the min-cut value is, the more vulnerable the local topology is.

Since the modification does not change the number of nodes in the input topology and the only difference is in the contraction sequence of the nodes, the computation complexity remains the same as that of the original min-cut algorithm, i.e., $O(n^4 \log n)$, if there are n nodes within k hops of the node computing the min-cut. If k is small, the complexity may be expected to be fairly low.

Each node periodically calculates its min-cut value and the size of the min-cut set based on its local topology, and the computed min-cut value and the size of the corresponding min-cut set are piggybacked onto the node's HELLO packet, which is periodically broadcasted to its neighbors. An R-node compares the min-cut values and the sizes of the min-cut sets of nodes in its k -hop local topology, and then it moves to the proximity of the particular normal node with the minimum min-cut value. If two normal nodes have the same min-cut value, then the R-node moves to the proximity of the node with a larger min-cut set.

The topology changes due to node mobility in MANETs. Accordingly, the R-nodes have to move to appropriate places in order to maintain the reliable routing framework. Some of the positions that the R-nodes choose may not be the optimal ones from the global point of view because each node just knows local topology and the change of the network makes it very difficult for the R-nodes to find the best positions.

In order to prevent multiple R-nodes from moving to the same location at the same time, an R-node sends out

a motion request to the normal node to which it intends to move. The R-node does not move until it receives a motion confirmation from that node. Some additional constraints could have also been incorporated, such as requiring that no two R-nodes can move into the close proximity of one another, and limiting the number of R-nodes within the range of a particular R-node.

The simulation studies of AODVM show that the min-cut based deployment strategy achieves a better performance than the degree based strategy. The degree based strategy can be classified into two kinds: degree based strategy I (R-nodes are placed in the proximity of nodes with the minimum degrees) and degree based strategy II (R-nodes are placed in the proximity of the highest-degree neighbors of these nodes).

Inspired by the idea of deploying R-nodes in AODVM, we improved the M-Zone protocol to make it applicable to more general MANETs.

3. Network Model

In M-Zone, we assume each node obtains its location by using GPS or other techniques, and the source knows the location of the destination via some location service, e.g. [15]. Each node maintains a k -hop vicinity routing table. The k -hop vicinity is the same as the k -hop local topology of AODVM. Holes inevitably exist in the large-scale mobile networks, and a small fraction of R-nodes which are the same as that of AODVM are placed in the networks. Each node knows its location and maintains k -hop vicinity information in M-Zone protocol, thus we use the min-cut based strategy in AODVM to deploy R-nodes.

Although each node knows its location and maintains a k -hop topology which AODVM uses to deploy R-nodes, AODVM does not utilize the information to discover routes because the route discovery of AODVM is reactive. M-Zone not only uses the information to deploy R-nodes, but also makes full use of them to discover routes. To discover N node-disjoint routes, the region between a source and a destination is divided into N strip-shaped zones based on geographic location in M-Zone.

Let the coordinates of the source and the destination be (x_1, y_1) and (x_2, y_2) respectively, and the straight line L between the source and the destination is given by the equation $Ax + By + C = 0$, $A = y_2 - y_1$, $B = x_1 - x_2$, and $C = x_2y_1 - y_2x_1$.

A node obtains its distance to L using the following Eq. (1):

$$D_i = (Ax + By + C) / \sqrt{A^2 + B^2} \quad (1)$$

where (x_i, y_i) denotes the location of the node. The distance can be negative from Eq. (1) in order to confirm which zone the node belongs to. A zone is a strip-shaped region bounded by two lines based on their distance to L and the zone width d is related to the values of k and N .

As shown in Fig. 1, for two paths, the ranges of the two zones are as follows: I $(-d, 0)$; II $(0, d)$.

For three paths, the ranges of the three zones are as

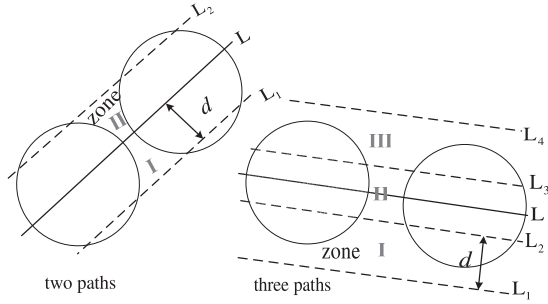


Fig. 1 The division of zones.

follows: I $(-3d/2, -d/2)$; II $(-d/2, d/2)$; III $(d/2, 3d/2)$.

We take the three paths case as an example to describe the method of zone division more clearly. Zone I is within the range $(-3d/2, -d/2)$, where $-3d/2$ is the distance from the boundary L_1 to L and $-d/2$ is the distance from the boundary L_2 to L . Zone II is within the range $(-d/2, d/2)$, where $-d/2$ is the distance from the boundary L_2 to L and $d/2$ is the distance from the boundary L_3 to L . Zone III is within the range $(d/2, 3d/2)$, where $d/2$ is the distance from the boundary L_3 to L and $3d/2$ is the distance from the boundary L_4 to L . Any node whose distance to L is within the range of a certain zone belongs to the corresponding zone. According to the two ways of dividing even and odd paths as shown in Fig. 1, we deduce that the boundaries of a zone can be described by the range $((-N/2 + j)d, (-N/2 + j + 1)d)$ where $(0 \leq j \leq N - 1)$. In this way, the lengths of the paths are not only close to each other, but also close to the shortest path length.

When $N > 2$, the zone width d has a threshold to find N paths through N zones. The transmission range of normal nodes is r . The k -hop vicinity is approximately a circle, the radius of which is kr . When N is odd, $d < (2kr)/(N - 2)$, and when N is even, $d < (2kr)/N$.

Since nodes can be mobile at will, we periodically recalculate the multiple zones according to the new locations of the source and the destination and it acts like that the multiple zones move periodically along with the movement of the nodes.

4. The M-Zone Protocol

To discover multiple node-disjoint routes, each route corresponds to a distinct zone respectively in M-Zone. In order to scale well in large-scale MANETs, the M-Zone protocol employs the segment-by-segment route discovery. We give a detailed description about the M-Zone protocol in this section.

4.1 Local Routing Table

Each node maintains a k -hop vicinity routing table in the network. If $k = 1$, it is location-based routing, and if the diameter of k -hop vicinity is approximate to the network diameter, it is proactive routing. Figure 2 illustrates a 2-hop vicinity of a node S . P belongs to this vicinity since the min-

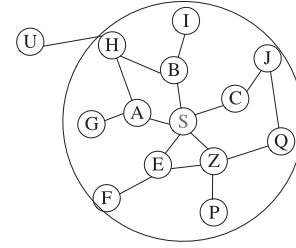


Fig. 2 A 2-hop vicinity of node S .

Table 1 Local routing table.

Destination	Next Hop List	Others
H	A, B	Hop, TTL, ...
G	A	Hop, TTL, ...

imum hop from S to P is 2. U is not in the 2-hop vicinity because the minimum hop from S to U is 3.

As R-nodes have more battery power, the transmission range and the k -hop vicinities of them are larger than that of normal nodes. Therefore, R-nodes can communicate with many more nodes than normal nodes.

There are two paths $S-A-H$ and $S-B-H$ from S to a destination H in Fig. 2, thus A and B are put into the next hop list in Table 1. In this way S can choose multiple paths to the nodes in its k -hop vicinity routing table to support discovering multipath routing in the whole network.

4.2 Route Discovery

When the destination is in the k -hop vicinity of the source, it uses proactive routing for route discovery. Otherwise, route discovery has three phases: the source to anchor phase, the anchor to anchor phase, and the anchor to destination phase. An anchor is the node nearest to the destination in the corresponding zone and within the k -hop vicinity of the previous anchor; the R-node which joins in the routes can also be called an anchor. Selecting anchors in this way ensures that the path length is as short as possible. The sub-path within the k -hop vicinity is called a segment and is obtained by using proactive routing.

When a normal node is chosen to forward packets, it will set a flag to indicate that it has been chosen. Then, when another node forwards packets to the node which are from the same source to the same destination, it will reply a packet such that it will not be chosen again. In fact, replying packets may happen in the phase when a segment goes through different zones. When a segment lies in a certain zone, a node will not receive repeat packets so that replying packets will not happen. In the source to anchor phase, the source chooses an anchor in each zone within the k -hop vicinity of the source. The N segments from the source to N anchors are built from the k -hop vicinity routing table maintained by the source. As for the anchor to anchor phase, each anchor chooses the next anchor in the same zone. The intermediate nodes in the segment from an anchor to the next anchor are selected according to the k -hop vicinity routing

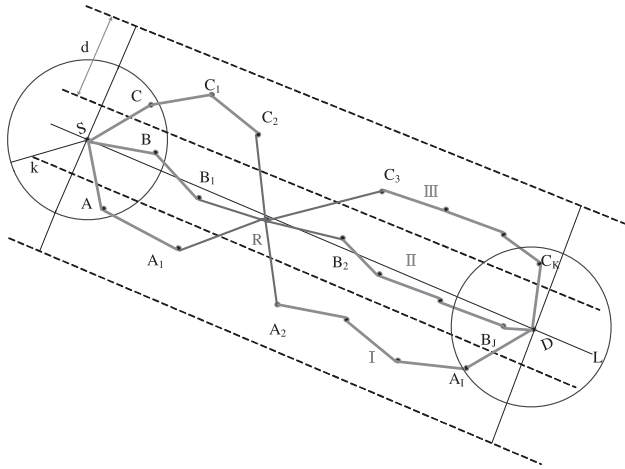


Fig. 3 Route discovery.

table of the former, and are located in the same zone as the anchor. This phase continues until the destination is within the k -hop vicinity of an anchor.

Finally, for the anchor to destination phase, the segment from an anchor to the destination can be computed according to the k -hop vicinity routing table of this anchor.

The routing is performed in this segment-by-segment manner, and the source can discover N node-disjoint routes to the destination through N zones. When there are holes in the network, it is difficult to find such multiple node-disjoint paths. Then normal nodes can forward packets to a reliable node nearby. We give an example to describe the route discovery in detail.

Figure 3 presents a route discovery procedure. A source S intends to forward packets to a destination D through three paths. S obtains the location of D by querying the location service, and includes it in the header of the packets. Here, D is not within the k -hop vicinity routing table of S . Thus, S calculates the straight line L according to the locations of S and D , and includes them in the header of the packets. Since S intends to find three paths, three zones need to be computed and included in the packets.

S broadcasts the packets in its k -hop vicinity, and the nodes involved calculate their own distance to L to confirm to which zone they belong. S determines three anchors A , B , and C respectively in three corresponding zones and then builds a segment to each anchor based on its k -hop vicinity routing table.

The segment from A to its next anchor A_1 is built according to the k -hop vicinity routing table of A and all the intermediate nodes in this segment belong to zone I. Anchors B and C determine their next anchors in the same way.

A_1 cannot find other appropriate normal nodes to forward packets but it finds that there is a reliable node R in its k -hop vicinity. A_1 forwards packets to R directly if R is the neighbor of A_1 ; otherwise, A_1 forwards the packets to R according to its k -hop vicinity routing table. The anchors B_1 and C_1 forward packets to R in the same way.

R finds three anchors A_2 , B_2 , and C_2 in the three corre-

sponding zones respectively in its k -hop vicinity. The three anchors can find their next anchors which are normal nodes within their k -hop vicinities respectively. Thus they carry out route discovery according to the former anchor to anchor way.

This procedure continues until the anchors A_I , B_J , and C_K find that D is in their k -hop vicinity routing tables. They build three segments to D according to their vicinity routing tables.

In this way, the three routing paths from S to D are finally built and they are node-disjoint due to separate zones and the mechanism of setting flags to sign forwarders.

There is not any route maintenance except for maintaining a k -hop vicinity routing table of each node in M-Zone when nodes move fast. The source initializes route discovery each time in the updated multiple zones based on the new locations of the source and the destination. It looks like that the multiple zones move according to the movement of the nodes. The particular node mobility seems to be hidden by these zones so that the route discovery appears simple and intuitive.

When nodes move slow in a network, it needs not to initialize route discovery every time in M-Zone. We employ the local route maintenance and global route maintenance to maintain the routes.

4.3 Route Maintenance

We use two approaches to maintain the routes: local route maintenance and global route maintenance.

In the local route maintenance, the source records its next anchors and next-to-next anchors, and the destination records its previous anchors and previous-to-previous anchors. Each anchor in the routing paths records its next anchor, next-to-next anchor, previous anchor, and previous-to-previous anchor.

When the movement of the source S results in its next anchor A moving out of its k -hop vicinity, S will select another node, which is in the same zone as A and is within the k -hop vicinity of both S and A , as its next anchor. Then A becomes the next anchor of the new anchor.

When an anchor C_1 moves out of the k -hop vicinity of its previous anchor C , C_1 will select a new anchor, which is in the same zone and within the k -hop vicinity of both C and C_2 , to replace C_1 .

When the movement of the destination D causes its previous anchor A_I moving out of its k -hop vicinity, A_I will choose another node whose k -hop vicinity contains the destination. This node will forward the packets to D .

If the M-Zone protocol uses local route maintenance for a long time, the path length will be greatly increased and the multiple zones will not be optimal. To resolve this problem, we use global route maintenance, which initializes route discovery periodically. Each node in the network maintains a k -hop vicinity routing table and the rediscovery time can be configured according to practical situations.

5. Performance Analysis

We firstly analyze the stability of the M-Zone protocol, and then we prove properties according to the following values: n and r denoting the number and the transmission range of normal nodes respectively; m and R denoting the number and the transmission range of R-nodes respectively; the number of paths N and the number of communication pairs e .

Node density in the network is ρ , which is defined as the average number of nodes residing in a unit area of one square meter. It can be computed as Eq. (2), where W is network width and L is network length.

$$\rho = \frac{m + n}{WL} \quad (2)$$

M-Zone has good stability because of the following reasons:

Firstly, if a path breaks, other paths can also be used to forward packets in other paths. The correlation factor among paths is lowest in node-disjoint multipath routing because the failure of a normal node will not cause other paths to break except for the path where it resides. As normal nodes are disjoint in M-Zone, the paths consisting of normal nodes are stable. Although R-nodes are the conjunctions with multiple normal nodes, these nodes will not fail easily because they are more reliable than normal nodes.

Secondly, each node maintains a k -hop vicinity routing table so that the segment is very stable. When nodes move fast, the source initializes route discovery every time such that the high mobility of nodes has little influence on the routing paths. When nodes move slow, the local routing maintenance and global routing maintenance are combined to maintain routes. Thus the M-Zone protocol has good stability no matter whether nodes move slow or fast.

Property 1: In the multiple paths between a source and a destination, the normal nodes are not common except for the source and the destination.

Proof: For a path, the anchors and the intermediate normal nodes between the anchors reside in a particular zone. These normal nodes in multiple paths belong to multiple zones respectively which are divided based on the geographic location. Normal nodes in different paths will not reside in the same zone at the same time. Then for different paths, the normal nodes will not be common.

A normal node will set a flag when it is chosen as a forwarder and then other nodes which forward packets from the same source to the same destination will not forward packets to it. The flag setting mechanism ensures that the normal nodes between a source and a destination will not be common further, especially for the normal nodes within the k -hop vicinities of the source and the destination. Therefore, the normal nodes are not common except for the source and the destination in the M-Zone protocol.

Property 2: The control complexity of M-Zone is lower than that of AODVM.

Proof: The control complexity is caused by the control packets. For M-Zone, the control overhead is caused mostly by maintaining the k -hop vicinity of each node. The motion request and motion confirmation messages of R-nodes are ignored here because they are much fewer than other control packets.

The k -hop vicinity is approximately a circle. For each normal node, the radius of its k -hop vicinity is $k \times r$, and the number of nodes within its k -hop vicinity is $\rho\pi(kr)^2$. Then it needs to forward control packets to $\rho\pi(kr)^2$ nodes once it updates the k -hop vicinity routing table. The whole normal nodes need to forward $n\rho\pi(kr)^2$ control packets when they update their k -hop vicinity routing tables. For each reliable node, the radius of its k -hop vicinity is $k \times R$, and the number of nodes within its k -hop vicinity is $\rho\pi(kR)^2$. Then it needs to forward control packets to $\rho\pi(kR)^2$ nodes once it updates the k -hop vicinity routing table. The whole reliable nodes need to forward $m\rho\pi(kR)^2$ control packets when they update their k -hop vicinity routing tables. In the k -hop vicinities of the source and the destination, the forwarder may reply packets. There are $2\rho\pi(kr)^2$ reply packets in the worst case. So the control complexity of M-Zone is $O(\rho\pi k^2(nr^2 + mR^2))$.

Except for the $\rho\pi k^2(nr^2 + mR^2)$ control packets to maintain k -hop local information, a large number of RREQ and RREP packets need to be forwarded in AODVM. As intermediate nodes cannot discard the duplicate RREQ packets, $(n\rho\pi r^2 + m\rho\pi R^2)$ RREQ packets need to be forwarded in the worst case. The number of RREP packets has relationship with the number of paths N . $(m + n)/N$ RREP packets need to be forwarded in the worst case.

Therefore, the control complexity of AODVM is $O(\rho\pi(nr^2 + mR^2)(k^2 + 1) + (m + n)/N)$, which is higher than that of M-Zone.

Property 3: The storage complexity of M-Zone is lower than that of AODVM.

Proof: The storage complexity measures the order of the routing table size used by the protocols [18]. For M-Zone and AODVM, each node maintains the k -hop vicinity information. The number of entries is $\rho\pi(kr)^2$ for a normal node and $\rho\pi(kR)^2$ for an R-node. The storage complexity of M-Zone is $O(\rho\pi(kR)^2)$.

In AODVM, the routing table needs to record the number of communication pairs except for the nodes in the k -hop vicinity. The storage complexity of AODVM is $O(\rho\pi(kR)^2 + e)$. Thus the storage complexity of M-Zone is lower than that of AODVM.

6. Simulation Studies

We simulate M-Zone and AODVM in the Mobility Framework model of OMNeT++. In the simulations, all nodes are distributed randomly but not uniformly such that there have sparse areas. The normal nodes move according to the random waypoint model, where the speed is uniformly and randomly distributed over $[0, 5 \text{ m/s}]$, and the transmission range is 250 m. For the R-nodes which account for 10% of all nodes, the moving speed is 20 m/s, and the transmission

range is 600 m. Each R-node compares the min-cut values and the sizes of min-cut value of nodes in its k -hop topology and moves to the proximity of the node with lowest min-cut value. Because the min-cut value represents the vulnerability of the k -hop topology, the R-node moves to the region which is most vulnerable based on the minimum min-cut value. In other words, the R-node moves to a sparse region. If the minimum min-cut value is the same in two or more than two normal nodes, then the R-node moves to the proximity of the node with a larger min-cut set.

We consider two network sizes $1,500 \text{ m} \times 1,500 \text{ m}$ and $4,000 \text{ m} \times 4,000 \text{ m}$, and the simulation time is 400 s. The following three performance metrics are used in the simulations:

Relationship between zone width d and hop number k : Given k , it shows the minimum value of d that can make the average path length shortest.

Average end-to-end delay: The average end-to-end delay mainly includes buffer latency during route discovery, queuing delay, and propagation delay.

Routing overhead: The total number of control packets generated by all the nodes during the simulations.

The relationship between k and d is done in the network size $1,500 \text{ m} \times 1,500 \text{ m}$ with 100 nodes. Generally, as k increases, d also increases in Fig. 4. We also see that at higher values of k , i.e., $k = 4$ and 5 , the change in d is less than that at lower values. For the given network size and the transmission range in the simulations, the 4-hop vicinity is large enough, and thus the influence on d becomes less.

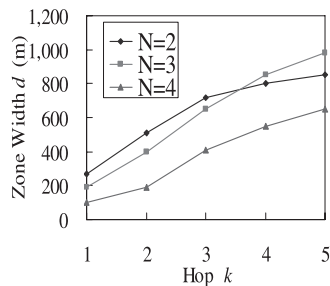


Fig. 4 The relationship between k and d .

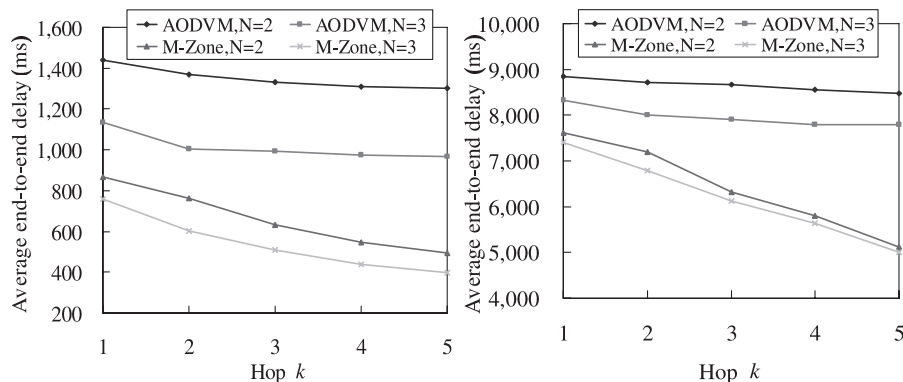


Fig. 5 Average end-to-end delay in two network sizes. Left: $2,000 \text{ m} \times 2,000 \text{ m}$; Right: $5,000 \text{ m} \times 5,000 \text{ m}$.

This means that the larger k is, the less influence k has on d . For a given k , d decreases as the node-disjoint path number N increases. However, when k equals to 4 or 5, the zone width d of $N = 3$ is larger than that of $N = 2$. There are two reasons for this result, one is that the change in d decreases as N decreases; and the other is that different methods of zone division are used, i.e., for odd and even paths which are described in detail in Sect. 3.

Figure 5 presents the results of average end-to-end delay at two network sizes. The value of k has little influence on the average end-to-end delay of AODVM, but as k increases, the average delay of AODVM presents a trend of descending. That is because more topology information can be available so that the R-nodes can obtain better position as k increases, thus the buffering latency during route discovery can be reduced. The value of k has more influence on the average end-to-end delay of M-Zone because the segments are built according to proactive routing. The larger k is, the more vicinity information can be available, and the buffering latency during route discovery can be reduced greatly.

For AODVM and M-Zone, load balancing can be achieved by spreading the traffic along multiple routes, which can alleviate congestion and bottlenecks. Therefore the average end-to-end delay decreases as N increases. The average end-to-end delay of M-Zone is lower than that of AODVM, because the data packets are forwarded after sending RREQ packets and receiving RREP packets in AODVM, which leads to long queuing delay. The data packets are forwarded directly according to local vicinity information and location information in M-Zone, so M-Zone takes advantages over AODVM in the average end-to-end delay.

Figure 6 presents the routing overhead at two network sizes. The routing overhead of both AODVM and M-Zone increases as k increases. For a given k , the routing overhead of M-Zone is less than that of AODVM. The reason is that the routing overhead is mostly caused by maintaining k -hop vicinity routing tables in M-Zone. But for AODVM, the routing overhead is caused not only by the k -hop vicinity maintenance which is used to deploy R-nodes, but also by a large number of RREQ packets which is very important to AODVM. The motion request and motion conformation

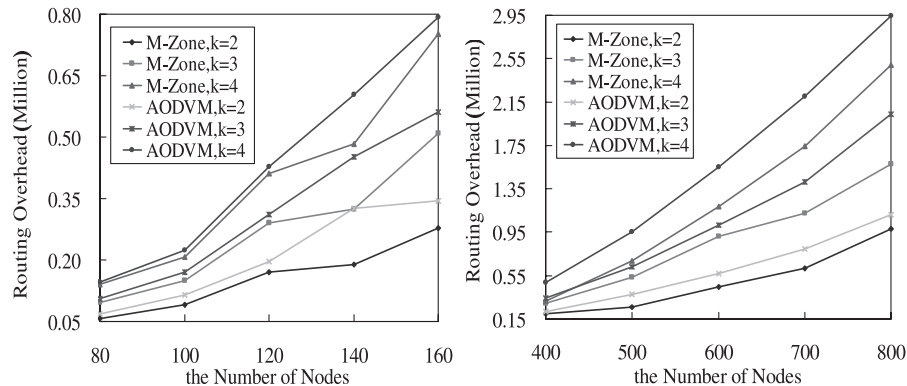


Fig. 6 Routing overhead in two network sizes. Left: 2,000 m × 2,000 m; Right: 5,000 m × 5,000 m.

messages of the reliable nodes cause little routing overhead compared with the large routing overhead.

It seems that the routing overhead for M-Zone ($k = 3$) and AODVM ($k = 2$) is the same when the number of nodes is 140. In fact the values are different since the interval value of the routing overhead 15,000 is large. The routing overhead increases slow when the number of nodes is between 120 and 140 for M-Zone as in Fig. 6 (Left). The node density is 120/(1,500 m × 1,500 m) when the number of nodes is 120, and 140/(1,500 m × 1,500 m) when the number of nodes is 140. The node density with the value 250/(2,500 m × 2,500 m) is moderate in Ye et al. [14], thus we believe that the node density is moderate when the number of nodes is between 120 and 140; and it achieves a better routing performance. But when the node density continues to increase, the node communication is disturbed and the number of the resending packets increases. As a result, the routing overhead increases obviously.

With the increase of the value of k , the average end-to-end delay of AODVM and M-Zone decreases, but the routing overhead of them increases obviously. How to select a moderate value of k is very important to improve the routing performance. The simulation studies show that M-Zone has obvious advantages in the 4,000 m × 4,000 m network size. M-Zone can be adaptive to large-scale MANETs better than AODVM because of using segment-by-segment routing discovery based on location information.

7. Conclusions

In this paper, we proposed the M-Zone protocol based on our proposed zoning method and also making use of the idea of deploying R-nodes in AODVM to find multiple normal node-disjoint paths in MANETs.

In the proposed protocol, the region between a source and a destination is divided into multiple zones, and each routing path goes through a distinct zone. The multiple zones move periodically as the source and the destination move, thus it seems that the mobility of particular nodes are hidden by multiple zones. Each segment within a vicinity is found according to proactive routing and anchors are selected according to location information. Therefore the M-

Zone protocol combines the advantages of proactive routing (short delay) and location-based routing (good scalability). The source initializes route discovery every time whenever nodes move fast. As the nodes move slowly, local route maintenance and global route maintenance are employed to maintain the routes.

The zone width is a constant value for multiple zones between a source and a destination, which is not good for finding routes. As our future work, we will try to find methods to make the zone width to be adaptive to network situations.

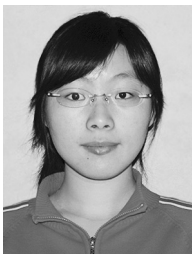
Acknowledgments

This work is supported by the National High-Tech Research and Development Plan of China (863 Plan) under Grant Nos. 2006AA01Z202 and 2006AA01Z199, the National Natural Science Foundation of China under Grant Nos. 60740440032 and 60533040, Shanghai Pujiang Plan No. 07pj14049, and the Program for Changjiang Scholars and Innovative Research Team in University under Grant Number IRT0661.

References

- [1] C.E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," Proc. ACM Conference on Communications Architectures, Protocols and Applications (SIGCOMM 1994), pp.234–244, Aug./Sept. 1994.
- [2] D. Johnson, D.A. Maltz, and J. Broch, "The dynamic source routing protocol for mobile ad hoc networks," Internet Draft, in Mobile Ad-hoc Network (MANET) Working Group, IETF, 1998.
- [3] C.E. Perkins and E.M. Royer, "Ad-hoc on-demand distance vector routing," Proc. IEEE Workshop on Mobile Computing Systems and Applications (WMCSA 1999), pp.90–100, New Orleans, Feb. 1999.
- [4] Z.J. Haas and M.R. Pearlman, "The zone routing protocol (ZRP) for ad hoc networks," Internet Draft, in Mobile Ad-hoc Network (MANET) Working Group, IETF, 1998.
- [5] L. Blazevic, S. Giordano, and J. Le Boudec, "Self organized terminode routing," Cluster Computing, Springer Science Business Media, pp.205–218, 2002.
- [6] J. Cao, L. Zhang, G. Wang, and H. Cheng, "SSR: Segment-by-segment routing in large-scale MANETs," Proc. Third IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS

- 2006), pp.216–225, Oct. 2006.
- [7] B. Karp and H.T. Kung, “GPSR: Greedy perimeter stateless routing for wireless networks,” *Proc. Annual International Conference on Mobile Computing and Networking (MobiCom 2000)*, pp.243–254, Aug. 2000.
 - [8] G. Wang, L. Zhang, and J. Cao, “Hole-shadowing routing in large-scale MANETs,” *International Journal of Sensor Networks (Inderscience Publishers)*, vol.4, no.4, pp.220–229, 2008.
 - [9] S. Mueller, R.P. Tsang, and D. Ghosal, “Multipath routing in mobile ad hoc networks: Issues and challenges,” *Lecture Notes in Computer Science (LNCS)*, vol.2965, pp.209–234, April 2004.
 - [10] S. Lee and M. Gerla, “Split multipath routing with maximally disjoint paths in ad hoc networks,” *Proc. IEEE International Conference on Communications (ICC 2001)*, vol.10, pp.3201–3205, June 2001.
 - [11] L. Wang, Y. Shu, M. Dong, L. Zhang, and O.W.W. Yang, “Adaptive multipath source routing in ad hoc networks,” *Proc. IEEE International Conference on Communications (ICC 2001)*, vol.3, pp.867–871, June 2001.
 - [12] X. Dong and A. Puri, “A DSDV-based multipath routing protocol for ad-hoc mobile networks,” *Proc. International Conference on Wireless Networks (ICWN 2002)*, 2002.
 - [13] M.K. Marina and S.R. Das, “On-demand multipath distance vector routing in ad hoc networks,” *Proc. IEEE International Conference on Network Protocols (ICNP 2001)*, pp.14–23, March 2001.
 - [14] Z. Ye, S.V. Krishnamurthy, and S.K. Tripathi, “A framework for reliable routing in mobile ad hoc networks,” *Proc. Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003)*, pp.270–280, April 2003.
 - [15] G. Wang, Y. Lin, and M. Guo, “A scalable and reliable multiple home regions based location service in mobile ad hoc networks,” *Proc. 2005 IFIP International Conference on Embedded and Ubiquitous Computing (EUC 2005)*, pp.550–559, Dec. 2005.
 - [16] R. Motwani and P. Raghavan, *Randomized Algorithms*, Cambridge University Press, Cambridge, 1995.
 - [17] Y. Ge, G. Wang, W. Jia, and Y. Xie, “Node-disjoint multipath routing with zoning method in MANETs,” *Proc. High Performance Computing and Communications (HPCC 2008)*, pp.452–460, Sept. 2008.
 - [18] X. Hong, K. Xu, and M. Gerla, “Scalable routing protocols for mobile ad hoc networks,” *IEEE Netw.*, vol.16, no.4, pp.11–21, 2002.

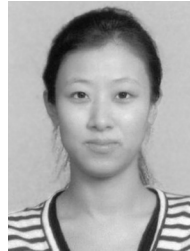


Yun Ge received B.Sc. in Computer Science in 2005 from Hefei University of Technology, China. She is currently a master student at School of Information Science and Engineering, Central South University, China. Her research interests include designing routing protocols in mobile ad-hoc networks.



during 2006–2007. His research interests include trusted computing, mobile computing, pervasive computing, and software engineering.

Guojun Wang received B.Sc. in Geophysics, M.Sc. in Computer Science, and Ph.D. in Computer Science, from Central South University, China. Since 2005 he is a Professor in Central South University. He is currently a Visiting Scholar in Florida Atlantic University and Adjunct Professor in Temple University, USA, during 2009–2010. He has been a Research Fellow in the Hong Kong Polytechnic University, Hong Kong, during 2003–2005, and a Visiting Researcher in the University of Aizu, Japan,



Qing Zhang received B.Sc. in Information Management and Information System in 2005 from Liaoning University of Technology, China. She is currently a master student at School of Information Engineering, Dalian Maritime University, China. Her research interests include designing routing protocols in mobile ad-hoc networks, grid computing and pervasive computing.



Minyi Guo received Ph.D. in Computer Science from the University of Tsukuba, Japan. Before 2000, he had been a research scientist of NEC Corp., Japan. He is now Chair and a Distinguished Professor at Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China. He was a professor at the School of Computer Science and Engineering, University of Aizu, Japan. He was also a visiting professor of Georgia State University, USA, Hong Kong Polytechnic University, University of Hong Kong, National Sun Yet-Sen University in Taiwan, China, University of Waterloo, Canada, and University of New South Wales, Australia. His research interests include parallel and distributed processing, parallelizing compilers, pervasive computing, embedded software optimization, molecular computing, and software engineering. He is a senior member of IEEE, a member of the ACM and IPSJ.