

GRID: A Fully Location-Aware Routing Protocol for Mobile Ad Hoc Networks

Wen-Hwa Liao, Yu-Chee Tseng, and Jang-Ping Sheu

Department of Computer Science and Information Engineering
National Central University
Chung-Li 32054, Taiwan
Email: whliao@axp1.csie.ncu.edu.tw
yctseng@csie.ncu.edu.tw
sheujp@csie.ncu.edu.tw

Abstract

A mobile ad hoc network (MANET) is one consisting of a set of mobile hosts capable of communicating with each other without the assistance of base stations. One prospective direction to assist routing in such an environment is to use location information provided by positioning devices such as global positioning systems (GPS). In this paper, we propose a new routing protocol called GRID, which tries to exploit location information in route discovery, packet relay, and route maintenance. Existing protocols, as compared to ours, are either not location-aware or partially location-aware in that location knowledge is not fully exploited in all these three aspects. One attractive feature of our protocol is its strong route maintenance capability — the intermediate hosts of a route can perform a “handoff” operation similar to that in cellular systems when it roams away to keep a route alive. This makes routes in the MANET more stable and insensitive to host mobility. Simulation results show that our GRID routing protocol can reduce the probability of route breakage, reduce the number of route discovery packets used, and lengthen routes’ lifetime.

Keywords: Global Positioning System (GPS), location-aware routing, mobile ad hoc network (MANET), mobile computing, routing, wireless network

Corresponding author: Yu-Chee Tseng

Department of Computer Science and Information Engineering
National Central University
Chung-Li 32054, Taiwan
E-mail: yctseng@csie.ncu.edu.tw
Phone: +886-3-422-7151 ext. 4512
Fax: +886-3-422-2681

1 Introduction

The advancement in wireless communication and economical, portable computing devices have made *mobile computing* possible [4]. One research issue that has attracted a lot of attention recently is the design of *mobile ad hoc network (MANET)*. A MANET is one consisting of a set of mobile hosts which can communicate with one another and roam around at their will. No base stations are supported in such an environment. Due to considerations such as radio power limitation, power consumption, and channel utilization, a mobile host may not be able to communicate directly with other hosts in a *single-hop* fashion. In this case, a *multi-hop* scenario occurs, where the packets sent by the source host are relayed by several intermediate hosts before reaching the destination host. Applications of MANETs occur in situations like battlefields or major disaster areas, where networks need to be deployed immediately but base stations or fixed network infrastructures are not available. A working group called “manet” has been formed by the Internet Engineering Task Force (IETF) to study the related issues and stimulate research in MANET [17].

Many routing protocols have been proposed for MANETs [1, 2, 5, 6, 8, 10, 20, 21, 22]. However, they are all based on a *graph model* in the sense that a mobile host only knows the connectivity relation with its neighbors, but not its relative location with its neighbors. Since data packets are actually routed in a physical area, protocols based on a *geographic model* have been proposed recently [9, 13, 14, 16]. It is assumed that a mobile host knows its current physical location, and thus such location information can be exploited to facilitate routing. We call protocols with such capability *location-aware* protocols. In [9], it is suggested to partition the physical area into a number of squares called *zones*. Routing is then performed in a two-level manner (through intra-zone and inter-zone). However, the protocol is a link-state one (or known as *proactive* protocol). Thus, any intra-zone link state change will be propagated to all other nodes in the zone, and any inter-zone link state change to the whole network. On the contrary, the LAR protocol proposed by [13] follows the *reactive* routing style, in that routes are searched in an on-demand manner. To save the route discovery cost, the current location of a source host and the past location of a destination host are used to confine the area to search for a route to the destination. In [16], no route search procedure is performed. Instead, to send a packet, a source node will forward the packet to the neighbor node that is closest to the physical location of the destination node. The same procedure is repeated until the destination node is reached, if possible. Several such approaches are discussed in [16]. In [14], a routing protocol is proposed for the *geocasting problem*, whose purpose is to deliver a message to the nodes currently resident within a specified geographical area; efforts were made to reduce multicast packets by limiting the forwarding space.

This paper investigates the routing problem in a MANET by exploiting the location information of mobile hosts. Basically, there are three issues to be addressed in this problem: *route discovery*, *packet relay*, and *route maintenance*. Depending on whether location information

Scheme	Route Discovery	Packet Relay	Route Maintenance
DSR [10]	no	no	no
AODV [21]	no	no	no
ZRP [6]	no	no	no
LAR [13]	yes	no	no
Lin [16]	*	yes	*
GRID	yes	yes	yes

Table 1: Comparison of routing protocols based on where location information is utilized to assist routing. * = does not perform this function.

is utilized in each of these issues, we classify routing protocols as *location-unaware*, *partially location-aware*, and *fully location-aware*. Our newly proposed protocol, called GRID, is fully location-aware because it tries to exploit location information in all these issues. A comparison of existing and our protocols based on such classification is in Table 1.

Our GRID protocol is a reactive protocol. Similar to [9], we treat the geographic area as a number of logical grids, each as a square. However, different from [9], in each grid, one mobile host (if any) will be elected as the leader of the grid. Routing is then performed in a grid-by-grid manner through grid leaders. One attractive feature of our protocol is its route maintenance capability. Given a route, as long as there exists a leader in each grid that constitutes the route, the route is considered alive. If a leader leaves its original grid, a behavior similar to the “handoff” procedure in cellular systems will take place. In this case, the leader will pass its routing table to the next leader (through broadcast). On the contrary, all other existing reactive protocols [10, 21, 13] treat a route as broken as long as *any* of the hosts on the route moves out of the transmission ranges of its neighbors. Thus, these protocols all suffer from short route lifetime when host mobility is high or when the route is long. With our protocol, these effects can be reduced significantly. In our protocol, even when the source and destination hosts roam away, a route may still function correctly. In addition, our protocol can eliminate the *broadcast storm problem*¹ that is associated with existing protocols when searching for a new route. In our protocol, because only grid leaders are responsible for route searching, the number of packets related to route search is insensitive to the density of mobile hosts in the searched area. On the contrary, all existing protocols will incur more control packets as the host density increases. Other possibilities that are investigated in this paper include different grid sizes and different strategies to confine the route search area. Through simulations, we justify that our GRID protocol will incur less route discovery packets, can reduce the probability of route breakage, and can lengthen the lifetime of routes.

The basic assumption in location-aware routing protocols is the availability of a positioning device such as a Global Positioning System (GPS) receiver at each mobile host [3, 12]. It

¹When searching for a route, typically a route request packet will be sent [1, 2, 5, 6, 8, 10, 13, 20, 21]. Every host in the searched area has the obligation to rebroadcast the packet. In [19], it is shown that serious redundancy, contention, and collision will be incurred in a MANET with such broadcasting.

is worth noting that GPS-related applications are quickly gaining popularity. As observed in [11, 15], location-aware or context-aware applications will be an important domain in mobile computing. Examples include navigation systems, telematic systems to facilitate communication with moving vehicles, geocasting, and tour guide systems. Availability of location information will have a broad impact on the application level as well as on the network level. Some new services such as geographic messaging, geographic advertising, and geographic services, are being considered [18].

The rest of the paper is organized as follows. Section 2 presents some background and motivation of this work. Our protocol is developed in Section 3. Section 4 shows how our protocol reduces the routing cost. Our experimental results are in Section 5. Section 6 concludes the paper.

2 Background and Motivation

2.1 Review of Routing Protocols for MANETs

A MANET consists of a number of mobile hosts which may occasionally communicate with one another. No base station is supported in the network. Each mobile host has a transceiver. Due to concerns such as transmission range and channel reuse, a mobile host may not be able to communicate with another host in a single-hop manner. In this, case, a *multi-hop* scenario occurs, where the packets of the source host are relayed by several intermediate hosts before reaching the destination host.

Many routing protocols have been proposed for MANETs [1, 2, 5, 6, 8, 10, 20, 21, 22]. Generally speaking, a routing protocol for MANET needs to address the following three issues:

- *Route Discovery*: When a mobile host wants to communicate with another mobile host, appropriate routing information has to be setup at the source and perhaps some intermediate nodes. According to how route information is collected, routing protocols can be classified as *proactive* and *reactive*. A proactive protocol attempts to continuously monitor the change of connectivity within the network, so that when a packet needs to be forwarded, a fresh route may exist immediately [20]. On the contrary, to reduce the effect of host mobility on routing, a reactive protocol invokes a route discovery/search procedure only when a route is needed. Several protocols are developed based on such on-demand concept [10, 21]. Reference [6] is a hybrid of proactive and reactive protocols.
- *Packet Relay*: This specifies how to forward data packets. Two ways are possible: *source routing* and *next-hop routing*. In source routing, the whole path to deliver a packet is specified in each packet header, and an intermediate node simply follows the path to deliver the packet (e.g., [10]). On the other hand, in next-hop routing, only the destination host is specified. Each intermediate node must keep a routing table to determine which node to forward the packet to (e.g., [21]).

- *Route Maintenance:* Due to reasons such as host mobility and interference, an established route may be broken. Route maintenance should concern with how route problems are reported and recovered. An established route may even be outperformed by a newly formed better route, and thus a route optimization may be executed [25].

2.2 Location-Aware Routing Protocols

Observe that a MANET is actually operating under a physical environment via radio signals. Routing could be greatly facilitated if locations of mobile hosts are known or can be approximated.

One location-aware routing protocol for MANETs is the LAR [13]. It tries to utilize location information in its route discovery procedure. For instance, in Fig. 1, when source host S needs a route to destination host D , the past location of D will be used to estimate the expected residential zone of D (shown in gray). Using the location of S and the expected zone, a rectangle (identified by corners A , B , C , and S) is constructed. This rectangle is used to confine the zone to be searched for a route from S to D . This will be better than a “blind” search and can save some route request packets. For instance, on receiving S ’s route request packet, host I will help to forward the request, but host J will ignore the request because it is out of the searched zone. Note that the success probability of the route discovery depends on the size and shape of the searched zone.

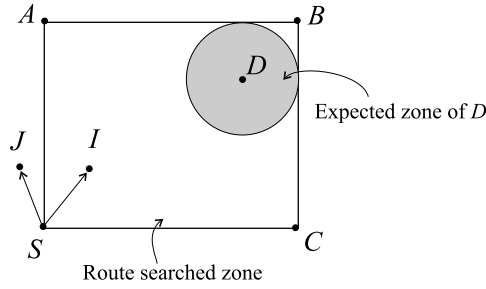


Fig. 1: Confining the searched zone in the LAR protocol.

Reference [9] proposes a zone-based two-level link state routing protocol. In this protocol, the physical area is partitioned into a number of squares called *zones*. Routing is performed in two levels: intra-zone and inter-zone. Two kinds of link states have to be maintained: the host connectivity information inside each zone and the zone connectivity information between adjacent zones. Inside a zone, each host periodically broadcasts its link connectivity information to its neighbors. From these information exchanges, each node will establish its intra-zone routing table containing routing information to nodes in the same zone. Thus, the intra-zone routing is a proactive one and connectivity information should be properly propagated every time when there is some change on link states, even if there is no need of performing communication. The proactive approach is generally considered less efficient than a reactive approach, as indicated by several recent papers [10, 21, 22]. In the inter-zone level, a node will

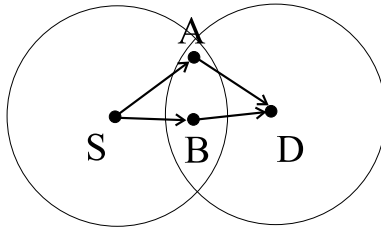


Fig. 2: Determining route quality based on location information.

periodically run an inter-zone clustering mechanism and broadcast a link-state packet when there is a change on its inter-zone routing table. This protocol has a communication overhead of $O(N^{3/2})$. Although the protocol takes a proactive approach, since the inter-zone routing table does not indicate where individual hosts reside in which zones, a location search is still needed before inter-zone routing can be performed. It is suggested to issue a unicast to each zone to inquire the existence of a destination host in that zone.

2.3 Observations and Motivations

The LAR protocol [13] only tries to use the location information to confine the searched zone to reduce the route discovery cost. To motivate our work, recall the three major issues of a routing protocol: route discovery, packet relay, and route maintenance. In the following, we make several observations to show how to further exploit location information in routing. (Note that the following observations only try to show more possibilities to exploit location information. Our yet-to-be-presented protocol does not necessarily implement each of these observations.)

First, in route discovery, the location information may be used to determine the quality of a route. For instance, in Fig. 2, we show two routes from node S to node D . Although both routes are of length 2, the one $S \rightarrow A \rightarrow D$ looks worse than $S \rightarrow B \rightarrow D$ because host A is located at the boundaries of both S 's and D 's transmission ranges. Not only this route has worse signal quality, but also it may experience higher chance of route breakage as A roams around.

Once a route discovery procedure is completed, the obtained routing information will be kept in some mobile hosts. Traditionally, routing information is recorded by identifying the *address* (e.g. IP address) of the next host to forward data packets. An interesting approach that will be explored in this paper is to identify the next hop by its *physical location*, instead of its address. Intuitively, when forwarding a data packet, a node may indicate that “Is there anyone around location L capable of forwarding the packet for me?” A routing protocol that can realize such concept will be more invulnerable to route breakage and may offer higher route quality.

For instance, in Fig. 3, consider the established route $A \rightarrow B \rightarrow C \rightarrow D$. When B roams off either A 's or C 's transmission range (in any of the directions shown in dotted arrows in the figure), this route will be considered broken if B must serve as an intermediate node. However,

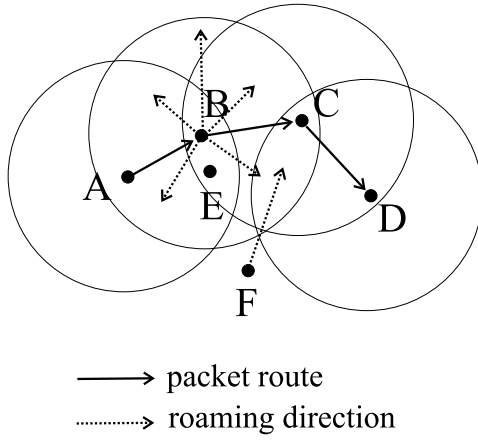


Fig. 3: Improving the vulnerability and quality of a route based on location information.

if B 's location is exploited, any host nearby B 's original location (e.g., host E in the figure) may work on behave of B to forward the packet for A to C . Thus, no new route discovery procedure needs to be initiated.

Such a feature will be very desirable in situations where the routing path is long and the host mobility is high, both of which will make a route more vulnerable to breakage. Furthermore, even if a route is not broken, location information can still help to improve the route's quality. For instance, in Fig. 3, when host F roams in the direction shown by the dotted arrow, using F as an intermediate node will be better than using C .

2.4 Positioning Systems

Here we briefly review some ways to identify the location of a device. The easiest way is perhaps through GPS, which is a worldwide, satellite-based radio navigation system [3, 12]. The system consists 24 satellites in six orbital planes operating in circular 10,900 nautical mile (20,200 km) orbits at an inclination angle of 55 degrees and with a 12-hour period. Operating on the L-band frequencies (1575.42 MHz and 1226.6 MHz), GPS can be used anywhere near the surface of the Earth. Satellites on the sky transmit navigation messages containing their orbital elements, clocks, and statuses, which can be used by a GPS receiver to determine its position and thus roaming velocity. Three satellites are necessary to determine the receiver's longitude and latitude, and four the receiver's altitude. More satellites can increase the accuracy of the readings, whose error typically ranges in a few tens of meters. To improve its accuracy, assistance from ground stations can be applied. Such systems, called *differential GPS (DGPS)*, can reduce the error to less than a few meters [15].

GPS receivers typically work well outdoors. For indoor location identification, short-range radios or infrared sensors can be used. An example is the Active Badge System developed by the Cambridge University [7, 24]. A badge is a small device that transmits a unique infrared signal every few seconds. A set of networked sensors throughout the site detects the badges' signals. The badge's location can be determined from the information provide by these sensors.

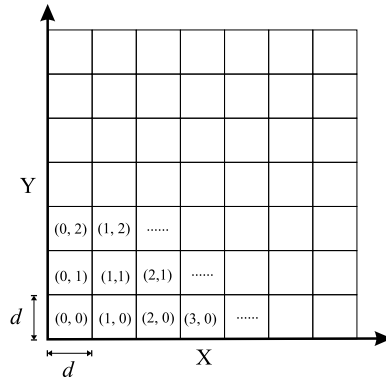


Fig. 4: Logical grids to partition a physical area.

3 The GRID Routing Protocol

3.1 Protocol Overview

Our protocol is called *GRID*. The geographic area of the MANET is partitioned into 2D logical grids as illustrated in Fig. 4. Each grid is a square area of size $d \times d$. Grids are numbered (x, y) following the conventional xy -coordinate. Each host still has a unique ID (such as IP address or MAC address). To be location-aware, each mobile host is equipped with a positioning device such as a GPS receiver from which it can read its current location. Given any physical location, there should be a predefined mapping from the location to its grid coordinate.

Routing is performed in a grid-by-grid manner. In each grid, one host will be elected as the *gateway* of the grid. The responsibility of gateway hosts includes: (i) forwarding route discovery requests to neighboring grids, (ii) propagating data packets to neighboring grids, and (iii) maintaining routes which pass the grid. All non-gateway hosts are not responsible for these jobs unless they are destinations of (i) and (ii) and sources/destinations of (iii). For maintaining the quality of routes, we also suggest that the gateway host of a grid should be the one nearest to the physical center of the grid.

One thing which is unspecified above, but will affect the performance of our protocol, is d (the side length of grids). Let r be the transmission distance of a radio signal. We discuss six possibilities of choosing d :

1. d is too large: The radio signal of a gateway host will have difficulty in reaching places outside of the grid, and thus a gateway-to-gateway communication is unlikely to succeed. So a d which is too large is unrealistic. (See Fig. 5(a), which shows the case of $d = 2r$.)
2. $d = r$: This represents the maximum value of d such that the gateways of two neighboring grids can talk to each other if they are located precisely at the centers of grids. (See Fig. 5(b).)
3. $d = \frac{2r}{\sqrt{10}}$: This represents the maximum value of d such that a gateway located at the center of a grid is capable of talking to any gateway of its 4 neighboring grids. (See Fig. 5(c).)

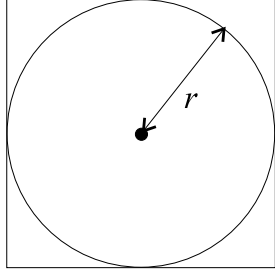
4. $d = \frac{\sqrt{2}r}{3}$: This represents the maximum value of d such that a gateway located at the center of a grid is capable of talking to any gateway of its 8 neighboring grids. (See Fig. 5(d).)
5. $d = \frac{r}{2\sqrt{2}}$: This represents the maximum value of d such that a gateway located at *any* position of a grid is capable of talking to any gateway of its 8 neighboring grids. (See Fig. 5(e).)
6. d is too small: This means that there will be very few, or sometimes no, mobile hosts resident in a grid. The chance of a mobile host becoming a gateway is high. In the extreme case, when d is infinitely small, there will be infinitely many grid and each host is the gateway of its own grid. In fact, this extreme case converges to the situation where there is no concept of grids, since each host will be responsible of forwarding route discovery and data packets. (See Fig. 5(f), which shows the case of $d = \frac{r}{10}$.)

The above discussion implies that a smaller value of d will lead to higher connectivity between neighboring grids. However, a smaller d also means a less number of hosts in a grid, which in turn implies a higher chance of a route being vulnerable to breakage (because once the gateway of a grid roams away, the chance that some other host will take over as the new gateway also becomes lower). So there exist some tradeoffs in choosing a good value of d (in Section 5, this issue will be studied).

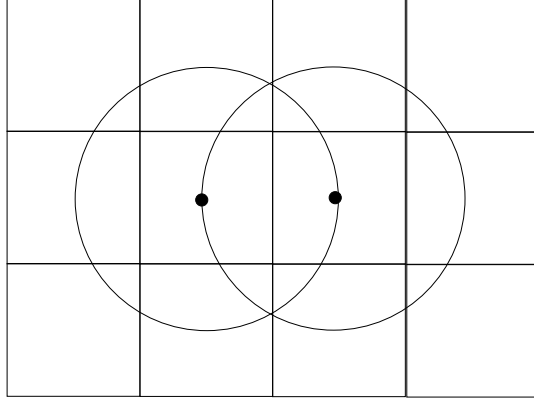
3.2 Route Search and Route Reply

This section discusses the route search and route reply procedures of our GRID protocol. It can be modified from any of the following protocols: source routing [10] and next-hop routing [21] (these protocols are location-unaware). The major differences are three-fold. First, we will use the locations of source and destination to confine the search range. Second, instead of letting every host to help with the search procedure, we only allow gateway hosts to take this responsibility. Third, the routing table will be set up in a grid-by-grid manner, instead of a host-by-host manner. In the following, we adopt the AODV protocol [21] (which is based on next-hop routing) to develop our protocol.

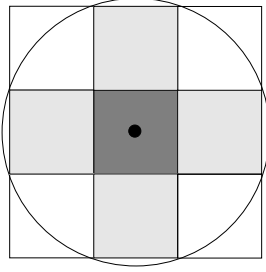
When a source node S at location l_S needs a route to a destination node D at estimated location l_D , it will broadcast a route request $RREQ(S, s_seq, D, d_seq, id, range)$ packet to request for a route to D . The pair (S, id) is to detect duplicate $RREQ$ packets from the same source S , so endless flooding of the same request can be avoided. The *source sequence number* s_seq is to specify freshness of a reverse route from the destination to the source, and the *destination sequence number* d_seq the freshness of a route from the source to the destination. The freshness information will be used to determine whether a route is acceptable or not. The parameter *range* is to specify an area to be searched for a route from S to D (to be discussed Section 4).



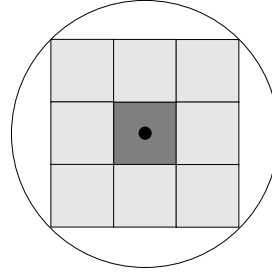
(a) $d = 2r$



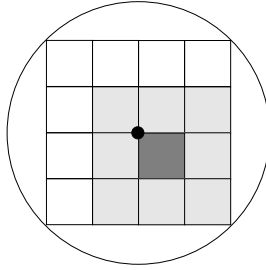
(b) $d = r$



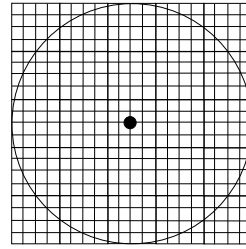
(c) $d = \frac{2r}{\sqrt{10}}$



(d) $d = \frac{\sqrt{2}r}{3}$



(e) $d = \frac{r}{2\sqrt{2}}$



(f) $d = \frac{r}{10}$

Fig. 5: The relationship between d (the side length of grids) and r (the radio transmission distance).

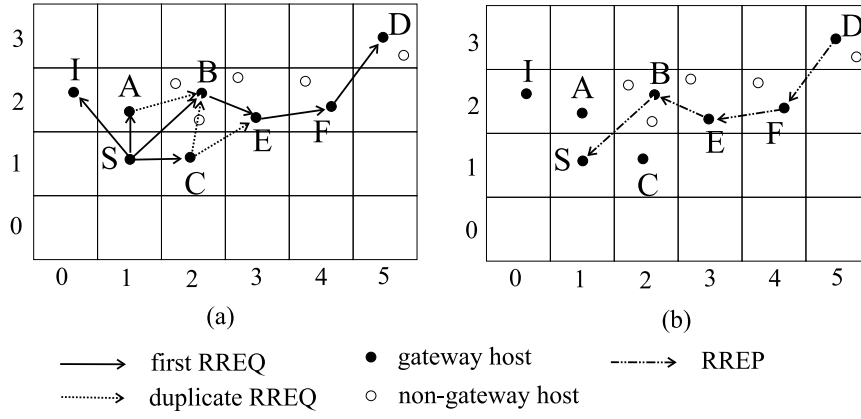


Fig. 6: An example of route discovery: (a) propagation of *RREQ* packets, and (b) propagation of *RREP* packets.

A gateway host, when receiving this *RREQ* packet, will first check whether it is located in the area specified in *range* or not. If so, it will set up a *reverse pointer* to the grid coordinate of the previous sending gateway and rebroadcast this packet. Otherwise, it simply ignores this packet.

When *D* receives the *RREQ*, a reply packet *RREP*(*S*, *D*, *d_seq*) will be sent from *D* to *S*. This packet is a unicast packet will travel following the reverse pointers which were established earlier when propagating *RREQ*. When a gateway receives the *RREP*, it will add an entry to its routing table indicating that there is a route from itself to *D* via the grid coordinate from which it received the *RREP* packet. When this *RREP* reaches host *S*, a route from *S* to *D* will be set up properly, on which data packets can be sent.

Let's use the example in Fig. 6 to show how the protocol works. Suppose $d = \frac{r}{2\sqrt{2}}$. First, a gateway election protocol is needed to elect a gateway in each grid (one such protocol is proposed in Section 3.3). Assume that hosts *A*, *B*, *C*, *E*, *F*, and *I* are the gateways of grids (1,2), (2, 2), (2,1), (3, 2), (4, 2), and (0, 2), respectively. Let host *S* be a source node initiating a *RREQ* packet to search for a route to destination *D* with the searched range confined by the rectangle formed by grids (1, 1), (5, 1), (5, 3), and (1, 3). When host *B* receives this *RREQ* for the first time, since it is within the searched range, it will rebroadcast this packet. Also, a reverse path pointing to grid (1, 1), where *S* resides, will be saved at *B*. Similarly, when *E* receives the *RREQ*, it will rebroadcast the packet and save a reverse path point to the previous grid (2, 2). The arrows in Fig. 6(a) show the *RREQ* packets progress. Finally, a reverse path from *D* to *S* will be established. Table 2 shows the reverse pointer at each intermediate host.

As the destination *D* receives the *RREQ*, it responds a *RREP* by unicast to *S*. This packet will follow the reverse path that was established by the *RREQ* packets. The progress of *RREP* is shown in Fig. 6(b). As the *RREP* packets traverse back to *S*, each gateway will establish an entry in its routing table showing the next grid leading to destination *D*, as shown in Table 3.

Node	S	B	E	F	D
Source	S	S	S	S	S
Request ID	1	1	1	1	1
Reverse Path	null	(1, 1)	(2, 2)	(3, 2)	(4, 2)

Table 2: The reverse pointer in each gateway established by the *RREQ* packets.

Node	S	B	E	F	D
Destination	D	D	D	D	D
Next hop	(2, 2)	(3, 2)	(4, 2)	(5, 3)	null

Table 3: The route entry in each gateway established by the *RREP* packets.

3.3 Route Maintenance

Next, we consider how to maintain a route once it has been established. The purpose is to keep the lifetime of a route as long as possible at the presence of host mobility. Under our protocol, except the source and destination hosts, each intermediate host must be a gateway. Therefore, the following two issues should be addressed: (i) how to maintain the gateway in each grid, and (ii) how to maintain a route when its source or destination node roams around.

To maintain the gateway in each grid, an efficient solution for gateway election is needed. We list the following guidelines in developing a good election protocol:

- When a new gateway should be elected, the mobile host nearest to the physical center of a grid should be selected. Such a host will be more stable because it is likely to remain in the grid for longer time. Thus, the election procedure will be executed less frequently and the protocol will be more bandwidth-efficient.
- To avoid the ping-pong effect, once a mobile host is elected as the gateway, it will remain so until it moves out of the grid. Thus, when another gateway roams closer to the physical center of the grid, it will not be elected as a gateway until the earlier one leaves the grid.

Now, we formally develop our gateway election protocol.

1. Periodically, a gateway host should broadcast its existence by sending a *GATE*(g, loc) packet, where g is its grid coordinate and loc is its current location.
2. Each mobile host should monitor the current gateway in its grid. If the *GATE* packet is not heard for a predefined time period, it will broadcast a *BID*(g, loc) packet, where g is its grid coordinate and loc is its current location. Upon the gateway host (if it is still alive and is in grid g) hearing the *BID* packet, it will reply a *GATE* packet to reject the former's bid. Upon a non-gateway at a location closer to the physical center of the grid hearing the *BID* packet, it will reply a *BID*(g, loc') packet to reject the former's bid, where loc' is the sending host's current location. If no such packets are received by the

bidding host for a predefined time period, the bidding host will silently elect itself as the current gateway without sending any packet (but it still has the obligation to announce its existence by following rule 1).

3. When a gateway host leaves its current grid, it should broadcast a *RETIRE*(g, T) packet, where g is the grid coordinate where it served as a gateway and T is the routing table at its hand. Every other host in this grid, on hearing this packet, will inherit the routing table T and take the same action as in rule 2 by sending *BID* packets to compete as a new gateway.
4. Each mobile host (including gateway and non-gateway) should monitor the existence of a gateway in each of its neighboring grids. When the mobile host roams into a new grid g in which it knows of no gateway existing, it will broadcast a *BID*(g, loc) packet to compete as a gateway, where loc is its current location. However, it is possible that multiple hosts will roam into grid g and initiate *BID* packets. If so, rule 2 will take action if some hosts disagree with this bidding.
5. To eliminate the possibility of having multiple gateways in a grid, when a host who assumes itself as the gateway hears the *GATE* packet from another host from a location closer to the physical center of its grid, it silently turns itself as a non-gateway without sending any packet.

Note that the last rule is necessary because broadcast is unreliable. Two *BID* packets may collide with each other without attention. However, note that here the gateway election does not have as rigorous constraints as those in the traditional *leader-election* problem [23]. Here, our protocol allows temporarily multiple gateways existing in a grid. If so, these gateways will all help with the route discovery procedure. Since we represent a route by grid coordinates instead of host ID's, this will not affect the correctness of our protocol.

The second problem is to solve the situation when the source or destination host roams off its original grid. It is desirable that under such situation, the route will not become broken. Let the source host roams from grid g to grid g' . We separate our discussion in four cases according to the roaming direction of the source node in the following:

1. Grid g' is the grid next to the source host on the route (see Fig. 7(a)). In this case, the route can still function correctly. Or, alternatively, the source node can consult with the gateway of g' (i.e., host B in the figure) to obtain the coordinate of the next grid on the route (i.e., host E) and then forward its data packets to that grid. In the latter case, the route will be one hop shorter.
2. The source node moves into a grid neighboring to grid g' (see Fig. 7(b)). In this case, no change is needed. Host S can still use its original routing table and forward its data packets to grid g' .

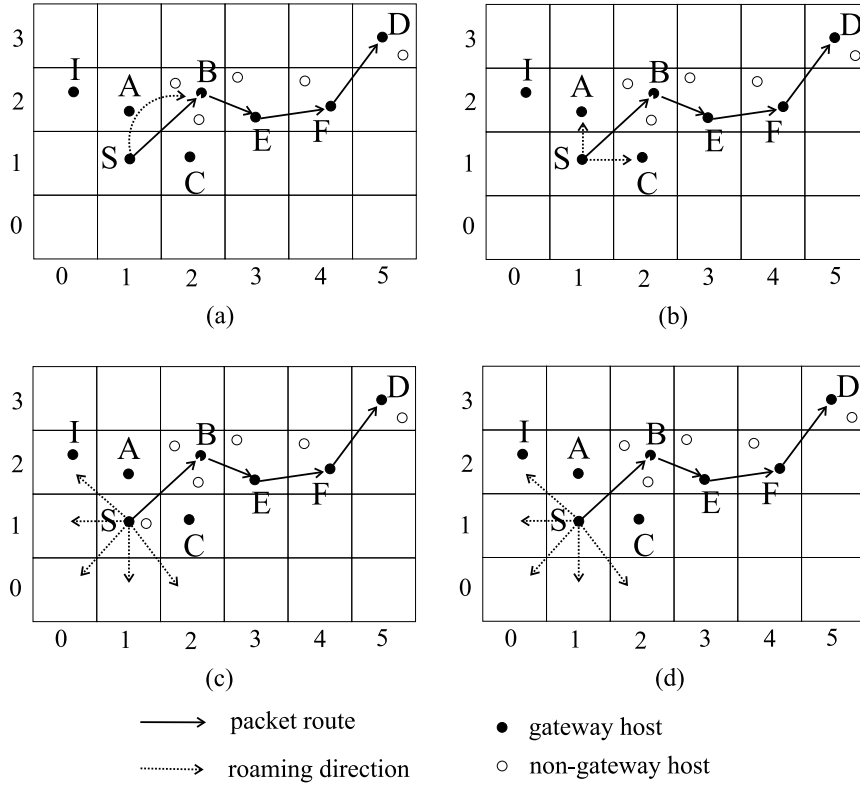


Fig. 7: Route maintenance when the source host roams off its current grid.

3. The source node moves into a grid not neighboring to grid g' and a gateway exists in grid g (see Fig. 7(c)). The source host will change its route entry for destination D to grid g . Then all data packets will be forwarded to grid g . In this case, the route is one hop longer.
4. The source node moves into a grid not neighboring to grid g' and a gateway does not exist in grid g (see Fig. 7(d)). This means there is no host to forward the source host's packets, and thus the route will be considered broken.

The rules for the movement of the destination node is similar to those for the movement of source node. We leave the development to the readers.

4 Eliminating the Broadcast Storm Effect

The route discovery procedure basically involves broadcasting a route request packet *RREQ* to other hosts. In the location-unaware protocols such as [1, 2, 8, 10, 21], this is done by a blind flooding. It is worth pointing out the result by [19], where it is shown that such a broadcast can easily lead to a storm effect causing serious redundancy, contention, and collision. First, because the radio propagation is omni-directional and a physical location may be covered by the transmission ranges of several hosts, many rebroadcasts are considered to be redundant. Second, heavy contention could exist because rebroadcasting hosts are probably close to each other. Third, collisions are more likely to occur because the RTS/CTS dialogue is inapplicable

and the timing of rebroadcasts is highly correlated. Collectively, these problems are called the *broadcast storm problem* [19].

In the LAR protocol [13], the route request packets will only be sent in a limited area. Thus, the broadcast storm problem can be alleviated to a certain degree. However, if the searched area is large, there will still be a lot of redundancy, contention, and collision in this area. This is also part of the reasons to motivate our work in this paper. In our GRID protocol, efforts are made in two directions to reduce the route search cost: to confine the route search range and to delegate the searching responsibility to the gateway hosts.

In our protocol, the parameter *range* in a *RREQ* packet is to confine the route searching area. Let S and D be the source and destination hosts, respectively. At host S , some records should be kept to identify host D 's recent location, roaming direction, roaming speed, etc. (say, through cache or other passing-by routing packets [13]). The freshness of these records should also be registered. Based on these records, host S will calculate the searched range. In the following, we suggest several ways to confine the searched area:

- *Rectangle*: the smallest rectangle that can cover the grid of S and the grid of D (see Fig. 8(a)). This is what is used in the LAR protocol [13].
- *Bar*(ω) : a bar from the grid of s to the grid of d with width ω (see Fig. 8(b)).
- *Fan*(θ, r) : a fan from the grid of s to the grid of d with angle θ and radius r (see Fig. 8(c)).
- *Two_Fan*(θ, r): the intersection of two fans, one from the grid of s to the grid of d , and the other from the grid of d to the grid of s , both with angle θ and radius r (see Fig. 8(d)).

Clearly, routes may fail to exist in the search area. In this case, a timeout mechanism should be installed so that when this happens, another round of route search can be initiated to search all areas for a route (we can let *range* = *NULL* to indicate this situation).

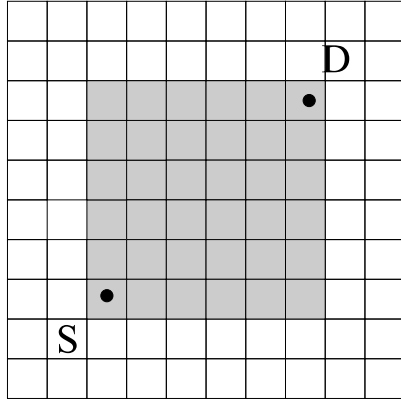
To see how many *RREQ* packets can be saved by our protocol, let the total number of hosts in the MANET be N . Using a blind flooding, the number of hosts that will try to broadcast the *RREQ* will be N (supposing that no hosts are isolated). After confining the route searched area, let A be the number of grids to be searched. The number of hosts that will try to broadcast the *RREQ* packet will be approximately

$$pA\delta + (1 - p)N,$$

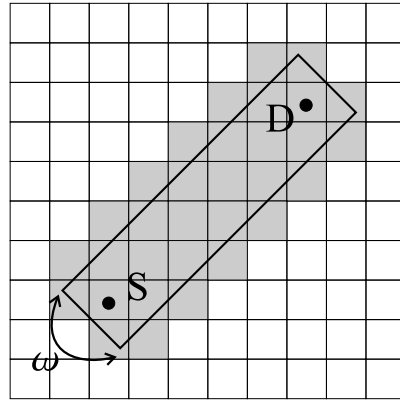
where p is the success probability of the first search and δ is the average number of hosts in a grid. Since our protocol will delegate the responsibility of route search to only the gateway hosts, the number of *RREQ* packets can be further reduced by about a factor of δ :

$$pA + (1 - p)N/\delta.$$

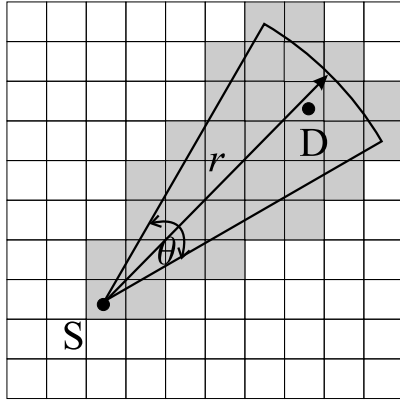
This leads to an attractive feature that the number of *RREQ* packets are insensitive to the host density.



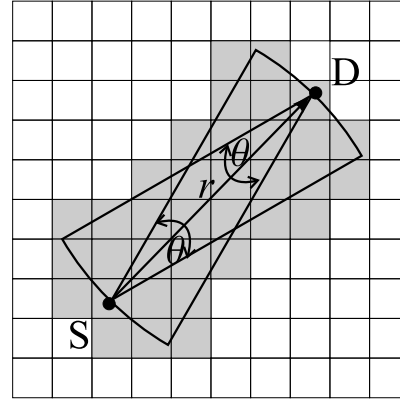
(a) Rectangle



(b) $\text{Bar}(\omega)$



(c) $\text{Fan}(\theta, r)$



(d) $\text{Two_Fan}(\theta, r)$

Fig. 8: Several ways to confine the route search area: (a): *Rectangle*, (b): *Bar*(ω), (c): *Fan*(θ, r), (d): *Two_Fan*(θ, r)

5 Experimental Results

5.1 The Simulation Model

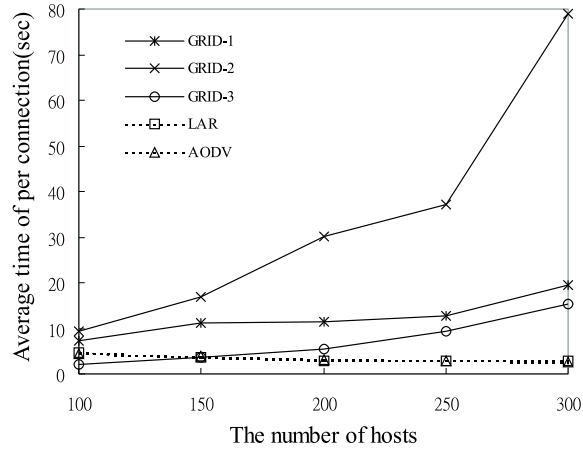
We have developed a simulator to evaluate and compare the performance of the AODV, LAR, and our GRID protocols. A MANET in a physical area of size $1000m \times 1000m$ with $100 \sim 300$ mobile hosts was simulated. A mobile host randomly chose its roaming direction every 0.5 second, and in each time two kinds of roaming speeds, 30 km/hr and 60 km/hr, were considered. All mobile hosts had the same transmission range of 300 meters. A gateway broadcasts its existence by sending a *GATE* packet every 10 seconds. Three values of grid size d were used (in decreasing order): $d = \frac{2r}{\sqrt{10}}$, $d = \frac{\sqrt{2}r}{3}$, and $d = \frac{r}{2\sqrt{2}}$ (called GRID-1, GRID-2, and GRID-3, respectively, in the following).

Each simulation run lasted for 500 seconds. Each simulated result was obtained from an average of 50 runs. In each run, 10 pairs of sources and destinations were selected randomly. Then data packets were generated at the source host by an exponential distribution with a mean of 0.1 sec throughout the simulation run. Whenever a data packet experienced a route breakage from the source to the destination, a route discovery procedure will be initiated. In both LAR and GRID protocols, we applied the Rectangle strategy in Section 3.3 to confine the searched zone. If this search failed, another round to search the whole area would be initiated. If again this failed, the search would be restarted after 2 seconds. The purpose was to avoid sending too many route request packets when the network was partitioned.

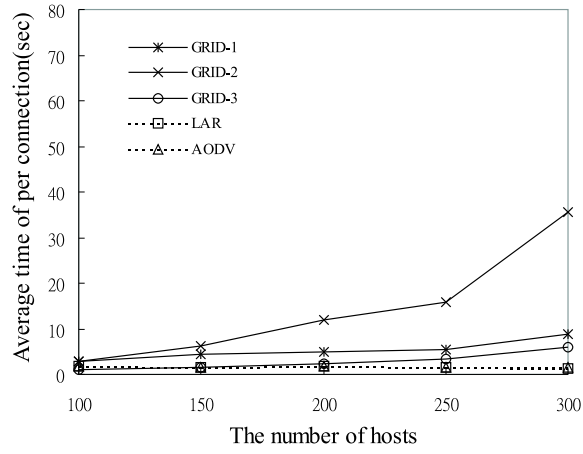
5.2 Observed Results

Fig. 9 shows the average lifetime of a route of each compared protocol. We vary the number of mobile hosts to observe the behavior. Since the physical area remains the same, a larger number of mobile hosts means a higher host density. Generally, GRID-2 has the highest route lifetime among all protocols, GRID-1 has the second highest route lifetime, and GRID-3 has the third highest route lifetime. This shows that the value of d should be chosen wisely to keep the connectivity between grids high while keeping a sufficient number of mobile hosts in each grid to reduce the possibility of route breakage. One attractive feature of our GRID protocol is that route lifetime will actually increase as the host density increases, while both the LAR and AODV protocols do not seem to benefit from this factor. Another observation is that when there are less than 150 mobile hosts, GRID-3 will have less route lifetime than LAR and AODV. The reason is that GRID-3 has a grid side length of $d = \frac{1}{2\sqrt{2}}r$, giving about a total number of 100 grids in the physical area. Thus, each grid will own about one mobile host, making maintaining a route alive more difficult. Also, as comparing Fig. 9(a) and Fig. 9(b), we see that a higher speed of mobile hosts will reduce route lifetime. Thus, it will be more important to use our GRID protocol to keep the route lifetime longer.

Next, we compare the bandwidth requirement of the route discovery procedure incurred by each protocol. We observe the total routing cost (i.e., route request, route reply, route error,

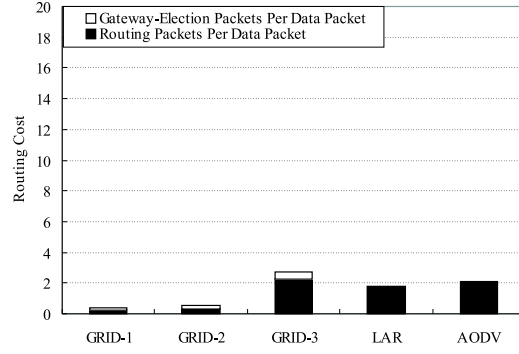


(a)

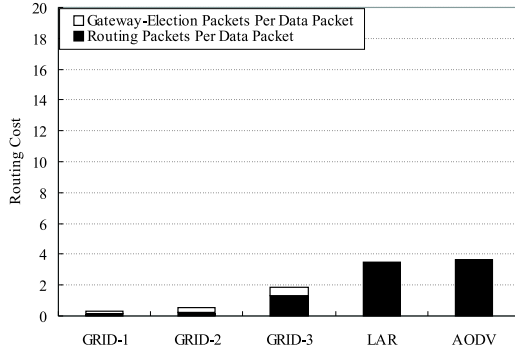


(b)

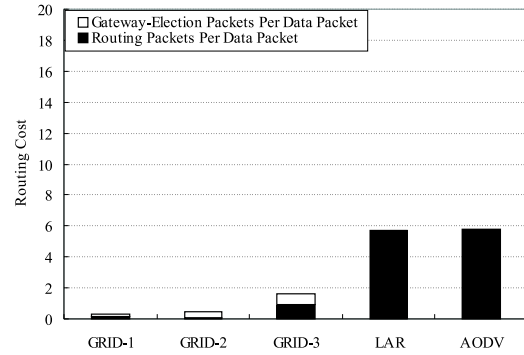
Fig. 9: Route lifetime vs. the total number of mobile hosts for protocols GRID-1, GRID-2, GRID-3, LAR and AODV: (a) roaming speed = 30 km/hr (b) roaming speed = 60 km/hr.



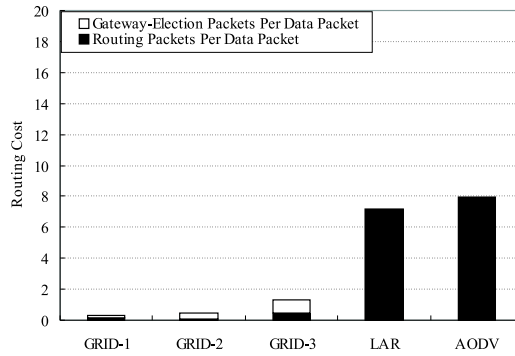
(a) Number of hosts = 100



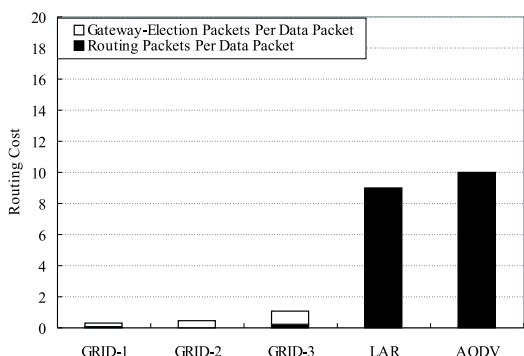
(b) Number of hosts = 150



(c) Number of hosts = 200

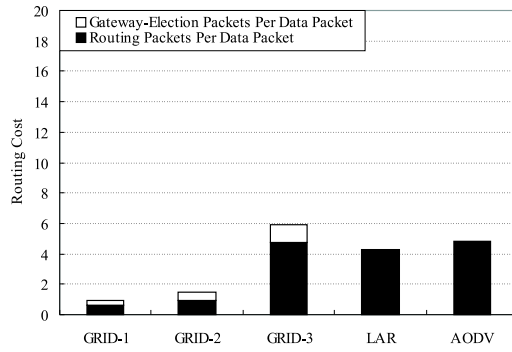


(d) Number of hosts = 250

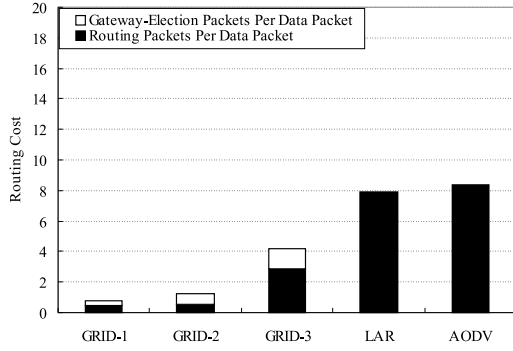


(e) Number of hosts = 300

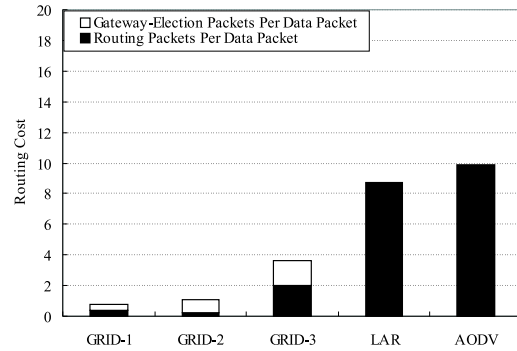
Fig. 10: Routing cost vs. the total number of mobile hosts for protocols GRID-1, GRID-2, GRID-3, LAR and AODV: roaming speed = 30 km/hr.



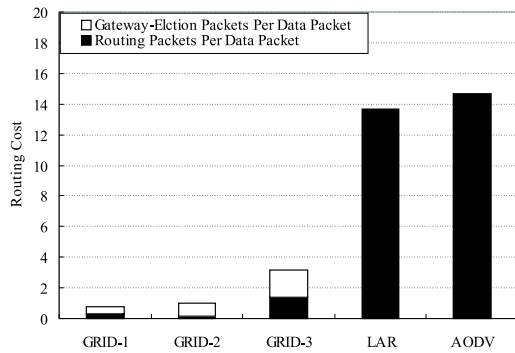
(a) Number of hosts = 100



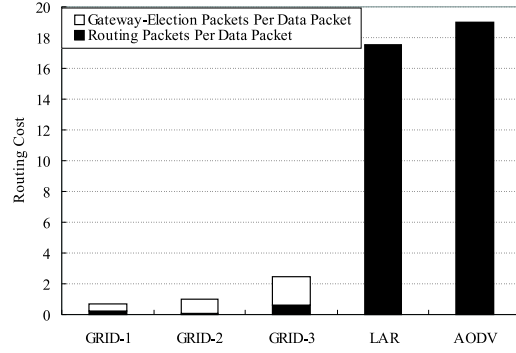
(b) Number of hosts = 150



(c) Number of hosts = 200



(d) Number of hosts = 250



(e) Number of hosts = 300

Fig. 11: Routing cost vs. the total number of mobile hosts for protocols GRID-1, GRID-2, GRID-3, LAR and AODV: roaming speed = 60 km/hr.

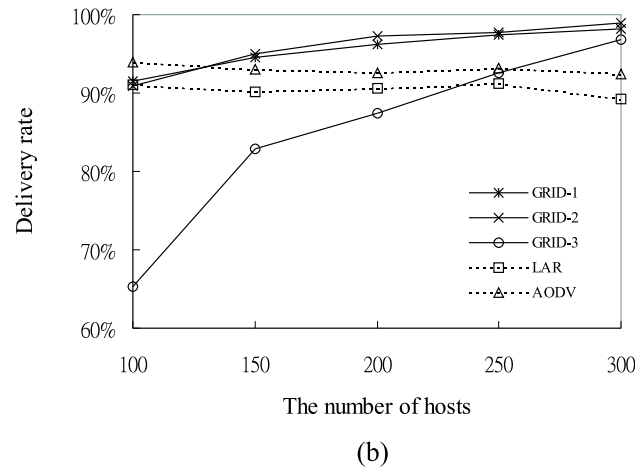
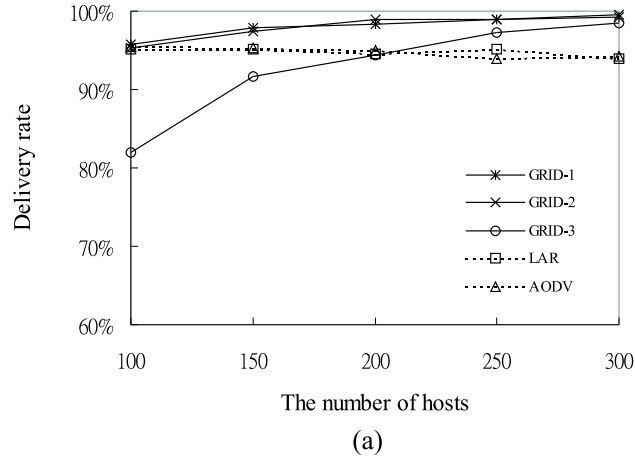
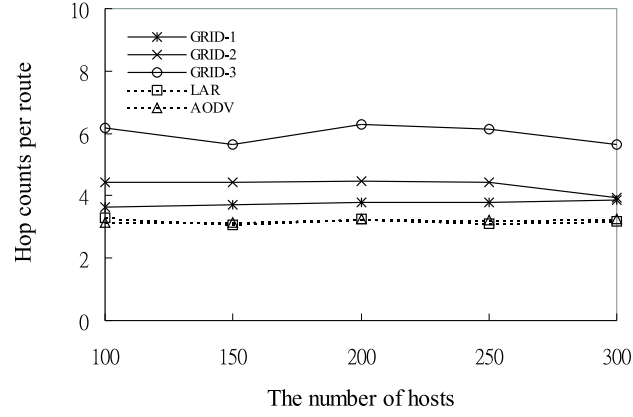
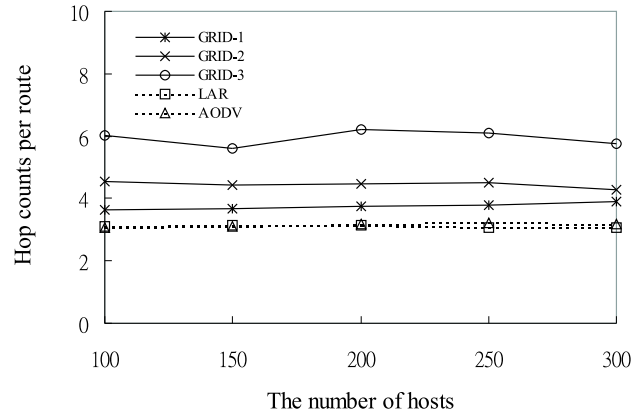


Fig. 12: Delivery rate vs. the total number of mobile hosts for protocols GRID-1, GRID-2, GRID-3, LAR and AODV: (a) roaming speed = 30 km/hr (b) roaming speed = 60 km/hr.



(a)



(b)

Fig. 13: Hop counts vs. the total number of mobile hosts for protocols GRID-1, GRID-2, GRID-3, LAR and AODV: (a) roaming speed = 30 km/hr (b) roaming speed = 60 km/hr.

and gateway election) incurred per delivered data packet. Note that the lost data packets are not taken into account in this calculation. The result is shown in Fig. 10 and Fig. 11. Generally speaking, GRID-1 and GRID-2 perform the best. Both LAR and AODV protocols have pretty high routing costs relative to our GRID-1 and GRID-2. As comparing GRID-3 with LAR and AODV, we find that GRID-3 is worse than LAR and AODV when the host density is low, but is better when the host density becomes higher. This is because GRID-3 uses a smaller d , leading to more *GATE* packets and more unreliable routes. Therefore, a significant gain can be offered by our GRID protocol, if the value of d is properly chosen.

From the above simulations, we can make some interesting observation on host density. When the host density increases, the routing costs of the LAR and AODV protocols will increase since there are more hosts trying to send route request packets. Note that although LAR uses location information to limit the route-searching range, the routing cost will still increase as host density increases. On the contrary, our costs slightly go down as the host density increases, since routes are becoming more stable with denser hosts. If we further observe the divide of gateway-election packets (*GATE*, *BID*, and *RETIRE*) and routing packets (route request, route reply, and route error), we see that the gateway-election packets are increasing with higher host density, but the routing packets are decreasing with higher host density. Overall, the decrease in routing packets is more significant than the increase of gateway-election packets.

We define *packet delivery rate* to be the number of data packets actually received by a destination divided by the number of packets issued by the corresponding source host. A comparison is in Fig. 12. We observe that GRID-1 and GRID-2 have the best delivery rates, especially when the host density is high. On the contrary, both LAR and AODV are quite insensitive to the host density. One special case is GRID-3, which has poor delivery rates with sparser hosts, and high rates with denser hosts. The reason is similar to the earlier discussion — there are too few hosts in each grid when the host density is low.

Finally, we examine the effect on route lengths by our GRID protocol. Fig. 13 shows the average route lengths of different protocols. LAR and AODV incur the lowest hop counts. Our GRID-1, GRID-2, and GRID-3 protocols will have about 0.5, 1, and 2.5 more hops, respectively. The reason is that our GRID protocol always confines relay hosts to gateway hosts, while LAR and AODV always try to search the route with the smallest hop counts. However, as shown earlier, using shortest routes is at the expense of less stable/reliable routing quality.

6 Conclusions

In this paper, we have presented a new location-aware routing protocol for MANETs. We have shown how to utilize location information to assist establishing a new route and maintaining existing routes in a MANET. The protocol is characterized by two nice features. First, it offers a much less routing cost than those of the existing protocols. This is achieved by confining the route searched zone to a limited area and by delegating the route search responsibility to

one mobile host (the gateway) of each grid area. Second, it offers much longer route lifetime than that of existing protocols. This is achieved by reelecting a new gateway after the previous gateway left its original location. To accomplish this, routing information in this paper is specified by *location* instead of *host address*. Since our protocol tries to forward packets in a grid-by-grid manner, as long as there exist a mobile host in each grid passed by a route, the route is considered alive. Therefore, our protocol is more resilient and less vulnerable to route breakage even under high host mobility. Simulation results do justify these benefits. Implementation is currently underway in the High-Speed Communication and Computation Laboratory at the National Central University. As pointed out by one referee, the gateway of a grid may not be able to communicate with some hosts inside the grid due to some obstacles. Thus, the grid may be logically partitioned into several sub-grids. This can be directed to a future research problem.

Acknowledgement

This research is supported in part by the Ministry of Education, ROC, under grant 89-H-FA07-1-4 (Learning Technology) and the National Science Council, ROC, under grant NSC89-2213-E-008-023.

References

- [1] J. Broch, D. B. Johnson, and D. A. Maltz. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (Internet draft). December, 1998.
- [2] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. In *Proc. of 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBILCOM '98)*, pages 85–97, 1998.
- [3] G. Dommety and R. Jain. Potential Networking Applications of Global Positioning Systems (GPS). In *Tech. Rep. TR-24, CS Dept., The Ohio State University*, April, 1996.
- [4] G. H. Forman and J. Zahorjan. The Challenges of Mobile Computing. *IEEE Computer*, 27(4):38–47, April, 1994.
- [5] M. Gerla and J. T.-C. Tsai. Muliccluster, Mobile, Multimedia Radio Network. *ACM/Baltzer Journal of Wireless Networks*, 1(3):255–265, July, 1995.
- [6] Z. J. Haas and M. R. Pearlman. The Zone Routing Protocol (ZRP) for Ad-Hoc Networks (Internet draft). August, 1998.
- [7] A. Harter and A. Hopper. A Distributed Location System for the Active Office. *IEEE Network*, 8(1), Jan., 1994.

- [8] M. Jiang, J. Li, and Y. C. Tay. Cluster Based Routing Protocol (CBRT) Functional Specification (Internet draft RFC). August, 1998.
- [9] M. Joa-Ng and I.-T. Lu. A Peer-to-Peer Zone-Based Two-Level Link State Routing for Mobile Ad Hoc Networks. *IEEE Journal on Selected Areas in Communications*, 17(8):1415–1425, 1999.
- [10] D. B. Johnson and D. A. Maltz. Dynamic Source Routing in Ad Hoc Wireless Networks. In *Mobile Computing, edited by Tomasz Imielinski and Hank Korth, Chapter 5, pages 153-181, Kluwer Academic Publishers*, 1996.
- [11] A. Jones. Mobile Computing to Go. *IEEE Concurrency*, 7(2):20–23, April-June, 1999.
- [12] E. D. Kaplan. *Understanding GPS: Principles and Applications*. Artech House, Boston, MA, 1996.
- [13] Y.-B. Ko and N. H. Vaidya. Location-Aided Routing (LAR) in Mobile Ad Hoc Networks. In *Proc. of the Fourth ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'98), Dallas, October, 1998*.
- [14] Y.-B. Ko and N. H. Vaidya. Geocasting in Mobile Ad Hoc Networks: Location-Based Multicast Algorithms. In *IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'99)*, February, 1999.
- [15] A. Krikelis. Location-Dependent Multimedia Computing. *IEEE Concurrency*, 7(2):13–15, April-June, 1999.
- [16] X. Lin and I. Stojmenovic. GEDIR: Loop-Free Location Based Routing in Wireless Networks. In *Proc. IASTED Int. Conf. on Parallel and Distributed Computing and System (to appear)*, 1999.
- [17] J. P. Macker and M. scott Corson. Mobile Ad Hoc Networking and the IETF. *ACM Mobile Computing and Communications Reviews*, 2(3):7–9, 1998.
- [18] J. C. Navas and T. Imielinski. Geographic Addressing and Routing. In *Proc. of the Third ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'97), Budapest, Hungary, September 26-30, 1997*.
- [19] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu. The Broadcast Storm Problem in a Mobile Ad Hoc Network. In *Proc. of the Fifth ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99)*, August, 1999.
- [20] C. Perkins and P. Bhagwat. Highly Dynamic Destination-Sequenced Distance-Vector (DSDV) Routing for Mobile Computers. In *ACM SIGCOMM Symposium on Communications, Architectures and Protocols*, pages 234–244, September, 1994.

- [21] C. Perkins and E. M. Royer. Ad Hoc On Demand Distance Vector (AODV) Routing (Internet draft). August, 1998.
- [22] E. M. Royer and C.-K. Toh. A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks. In *IEEE Personal Communications*, pages 46–55, April, 1999.
- [23] A. Silberschatz, J. Peterson, and P. Galvin. *Operating System Concepts*. Addison-Wesley, 1991.
- [24] R. Want, A. Hopper, V. Falcao, and J. Gibbons. The Active Badge Location System. *ACM Trans. on Information Systems*, 10(1):91–102, January, 1992.
- [25] S.-L. Wu, T.-K. Lin, Y.-C. Tseng, and J.-P. Sheu. Route Optimization on Wireless Mobile Ad-Hoc Networks. In *The 5th Mobile Computing Workshop, Taiwan*, pages 143–150, 1999.