Energy-Conserving Grid Routing Protocol in Mobile Ad Hoc Networks*

Chih-Min Chao, Jang-Ping Sheu, and Cheng-Ta Hu
Department of Computer Science and Information Engineering
National Central University, Chung-Li 320, Taiwan

Abstract

The lifetime of a mobile ad hoc network (MANET) depends on the durability of the battery resource of the mobile hosts. Earlier research has proposed several routing protocols specifically on MANET, but most studies have not focused on the limitations of battery resource. This study proposes a new energy-aware routing protocol, which can increase the durability of the energy resource and, therefore, the lifetime of the mobile hosts and the MANET. The proposed protocol can conserve energy by shortening the idle period of the mobile hosts without increasing the probability of packet loss or reducing routing fidelity. Simulation results indicate that this new energy-conserving protocol can extend the lifetime of a MANET.

Keywords: Ad hoc networks, energy-conserving, grid, location-aware, wireless communications.

1 Introduction

A mobile *ad-hoc network (MANET)* is formed by a cluster of mobile hosts without any pre-designed infrastructure of the base stations. A host in a MANET can roam and communicate with other hosts, at will. Two mobile hosts may communicate with each other either directly (if they are close enough) or indirectly, through intermediate mobile hosts that relay their packets, because of transmission power limitations. A main advantage of a MANET is that it can be rapidly deployed since no base station or fixed network infrastructure is required. MANETs can be applied where pre-deployment of network infrastructure is difficult or impossible (for example, in fleets on the oceans, armies on the march, natural disasters, battle fields, festival grounds, and historic sites).

Many routing protocols have been proposed for MANET [1, 2, 3, 4]. Most of them concentrate on the issues like the packet deliver ratio, routing overhead, or shortest path between source and destination. In fact, energy-constraints represent an equally important issue in MANET operations. Each mobile host that operates in a MANET has a limited lifetime due to its limited battery energy. Failure of one mobile host may disturb the whole MANET. Thus, battery energy should be considered to be a scarce resource and an effective energy-conserving technique must be found to extend the lifetime of a mobile host and, hence, the whole MANET.

Several studies have addressed energy-constraints. In [5], a minimum-power tree is established from source to destination to support broadcast/multicast services. Rodoplu and Meng [6] proposed a distributed power-efficient transmission protocol to reduce the power consumption of a mobile host and thus increase the lifetime of the whole network. In [7], the topology of the whole network is controlled by adjusting the transmission power of mobile hosts. The goal is to maintain a connected network using minimum power. Wu, Tseng, and Sheu [8] proposed an energy-efficient MAC protocol to increase channel utilization and reduce both power consumption and co-channel interference.

Besides reducing transmission power, the energy of a mobile host can be conserved by occasionally turning off its transceiver [9, 10]. As is well known, much power consumption is consumed during transmission and reception by a mobile host. However, if the transceiver is powered on, then the power consumption is not reduced much even through the mobile host is idle [11].

A mobile host still consumes much energy even when idle. Turning off the transceiver and entering sleep mode whenever a mobile host is neither transmitting nor receiving, is a better way to conserve energy. The problem with turning off the transceiver is that the mobile hosts may fail to receive packets. Two issues should be addressed in this energy conservation problem - (1) when should the transceiver be turned off and (2) how can packet loss be avoided when the destination host is in sleep mode. A longer sleep is preferred to conserve energy. That is, the transceiver should be turned off as soon as possible when idle. However, long sleeping increases the probability of losing packets. This work considers these two issues together, and seeks to maximize energy conservation without increasing the probability of packet loss.

The proposed protocol, *Energy-Conserving GRID (EC-GRID)*, exploits the concept of a routing protocol called *GRID* [2] while considering the energy-constraints. In GRID, each mobile host has a positioning device such as a Global Positioning System (GPS) receiver to collect its current position. The geographic area of the entire MANET is partitioned into 2D logical *grid*. Routing is performed in a grid-by-grid manner. One mobile host will be elected as the gateway for each grid. This gateway is responsible for (1) forwarding route discovery requests to neighboring grids, (2) propagating data packets to neighboring grids, and

---

*This work was supported by the National Science Council, Republic of China under the grant number NSC 91-2213-E-008-027.

0190-3918/03 $ 17.00 © 2003 IEEE

(3) maintaining routes for each entry and exit of a host in the grid. No non-gateway hosts are responsible for these jobs unless they are sources/destinations of the packets. 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.

In ECGRID, grid partitioning is the same as in the GRID routing protocol. The main difference between these two protocols is that ECGRID considers the energy of mobile hosts but the GRID does not. For each grid, one mobile host will be elected as the gateway and others can go into sleep mode. The gateway host is responsible for forwarding routing information and propagating data packets as in GRID. Sleeping non-gateway hosts will return to active mode by the signaling of the gateway, whenever data have been sent to them. (Accordingly, the transceiver is not periodically restarted to check whether data are to be received.)

The goal of this work is similar to that of Span [10] and GAF [9]. In Span, each mobile host switches between the coordinator and non-coordinator, according to a "coordinator eligibility rule". Span coordinators stay awake continuously to perform packet routing. Span non-coordinators stay in sleep mode and wake up periodically to check whether any packets have been sent to them. In GAF, the geographic area is partitioned into grids as in GRID, and hosts in the same grid are defined as routing equivalent hosts. In a grid, one mobile host is active and others can sleep. A host will set the sleeping duration before it goes to sleep. After this period of sleeping, the host will wake up to check its activity.

ECGRID, Span, and GAF are compared as follows. ECGRID is superior since hosts need not periodically wake up from the energy-saving state, unlike in the other two schemes. In ECGRID, hosts can be awakened by a signal from the gateway host whenever packets must be sent to them. This signaling ensures that the probability of packet loss will not increase because of the power saving operations of ECGRID. In Span, an ATIM (Ad Hoc Traffic Indication Map) mechanism is proposed to solve this problem. GAF includes no way to ensure that a destination host is active when packets are sent to it. Thus the packets cannot be delivered to the sleeping destination. In a location-aware scheme, such as ECGRID or GAF, more energy can be saved when host density is higher because only one host (gateway) in a grid is active. As the grid contains more hosts, each host can take turns to act as the gateway. Thus the saved power is proportional to host density. On the contrary, Span (not location-aware) does not benefit from increasing host density [10].

2 System Environment

The MANET is partitioned into 2D logical grids. This is exactly the same partition method as described in [2]. Each grid is a square area of size $d \times d$. Grids are numbered (x, y) following the conventional (x, y)-coordinate. Each host still has a unique ID (such as IP address or MAC address). Each mobile host is made location-aware by being equipped with a positioning device, such as a GPS receiver, from which it can read its current location. Then we can easily map the location information into its grid coordinate.

As mentioned above, each grid is a square of d x d. Let r be the transmission distance of a radio signal. In this study, the value of d is chosen as $\sqrt{2}r/3$. This value setting means that a gateway located at the center of a grid can communicate with any gateway in its eight neighboring grids.

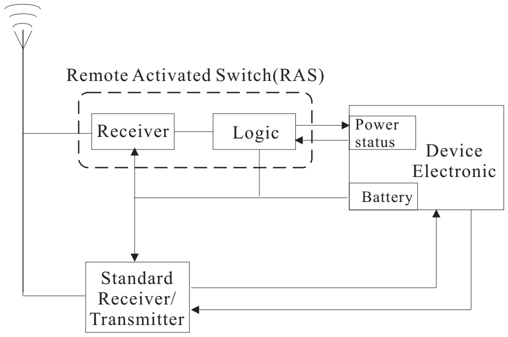


Figure 1. **The architecture of mobile host.**

The mobile hosts are assumed to have the same power supply. That is, all the mobile hosts have the same maximum energy. Each mobile host periodically calculates its *ratio of battery remaining capacity* (R_{brc}). R_{brc} is defined as,

$$R_{brc} = \frac{Battery\ remaining\ capacity}{Battery\ full\ capacity} \qquad (1)$$

Battery remaining capacity represents a mobile host's remaining energy while *Battery full capacity* means a host's maximum energy. Three energy levels are defined for each mobile host, according to this R_{brc}: *upper level* if $R_{brc} \geq$ 0.6; *boundary level* if $0.2 \leq R_{brc} < 0.6$, and *lower level* if $R_{brc} < 0.2$. This classification of remaining energy is used in the *gateway election rules*, which will be presented in next section.

Each host in a MANET is either in active mode. or in sleep mode A mobile host in active mode can transmit and receive packets. Only one host (the gateway) is active in each grid. Other non-gateway hosts can turn their transceivers off and enter sleep mode if they do not have packets to transmit or receive. These non-gateway hosts need not periodically wake up to check whether any pending packet is at the gateway. Instead, whenever the gateway receives a packet that is sent to a host in sleep mode, the gateway will actively wake the host up. This method is made feasible by equipping every host with a device called a Remotely Activated Switch (RAS) [12], as depicted in Fig. 1. As its name reveals, an RAS module can be used remotely to activate a mobile host. This RAS is fulfilled by *radio frequency tags* technology. When RAS receives a correct *paging sequence* (a set of paging signals), it can turn on the transceiver and bring the host into active mode. Thus, if a unique paging sequence is assigned to each host, any host can be awakened by sending its paging sequence. Note that the power consumption of RAS is much lower than the transmitting/receiving power consumption, and can thus be ignored in energy calculations.

3 The Energy-Conserving Grid Routing Protocol

A routing protocol called *Energy-Conserving GRID (ECGRID)* is proposed. Every mobile host in the network

0190-3918/03 $ 17.00 © 2003 IEEE

must run ECGRID. Each host uses its unique ID as the paging sequence. One mobile host in each grid will be elected as the gateway. The gateway host must maintain a *host table* that stores the host ID and status (transmit/sleep mode) of all the hosts in the same grid. Whenever the gateway receives a packet sent to a sleeping host, the gateway can wake up the mobile host by sending the paging sequence associated with that host.

A gateway may initiate a new gateway election process when it is leaving the grid, or seeking to maintain the load balance. To elect a new gateway, all hosts in the same grid must be in active mode. Each grid has a unique "*broadcast sequence*", which is defined as the *coordinate* of the grid. All hosts must move into active mode when they hear the broadcast sequence. That is, a gateway can wake up all the hosts in the same grid by sending the broadcast sequence. And then, a new gateway can be elected according to the gateway election rules. This new gateway will inherit the routing table from the original gateway and remains continuously active. In the meanwhile, the new gateway will inform its neighbors of the gateway change.

The gateway is responsible for forwarding routing information and transmitting data packets. It plays the most important role in our protocol and consumes much energy. All hosts should take their turn as the gateway to prolong the lifetime of the network. The election of the gateway should take the remaining battery capacity and position into consideration. A gateway with most remaining energy and nearer the center of a grid is preferred. These two principles prevent another gateway election from being triggered soon. The gateway election rules are as follows.

Gateway election rules:

1. A host with higher level of battery remaining capacity has higher priority
2. Given several hosts with the most energy level, the one that is closest to the center of the grid will be elected as the gateway. This rule allows a host that will stay in the grid for longer, be the gateway.
3. If no gateway can be elected according to steps 1 and 2, then the host with smallest ID (IP address or MAC address) will be elected as the gateway.

3.1 Gateway Election

The gateway election algorithm is executed distributively whenever a new gateway is needed, such as when the network is first initialized or when the gateway host runs out of energy, moves out of the grid, or turns down because of an accident.

Gateway election algorithm:

1. Each host in active mode will periodically broadcast its $HELLO$ message. The $HELLO$ message contains the following five fields.
    (a) id : host ID
    (b) $grid$: grid coordinate
    (c) $gflag$: gateway flag (set to 1 when the host is the gateway)
    (d) $level$: remaining battery capacity level (upper, boundary, or lower)
    (e) $dist$: distance to geographic center of the grid
2. After a $HELLO$ period, which is predefined as the period for hosts to exchange their $HELLO$ messages, all hosts are supposed to receive the $HELLO$ messages from neighboring hosts in the same grid. Then, each host will apply the gateway election rules to decide whether it becomes the gateway.
3. The host will declare itself as the gateway by sending a $HELLO$ message with the $gflag$ set. The gateway host is responsible for maintaining the host table, which is constructed from the id field of the $HELLO$ messages.
4. All other non-gateway hosts, receiving the $HELLO$ message from the gateway, will move into sleep mode if they have no packets to transmit.

3.2 Gateway Maintenance

The correct operation of the gateway is critical in the protocol. In the following, two aspects of the selection of the gateway are discussed - the *mobility of mobile hosts* and the *load balance of mobile host's battery energy*.

Before entering sleep mode, each mobile host will set a timer to wake up. This timer is set to the estimated dwell duration over which the host is expected to remain in its current grid. The estimation depends on the location and velocity of the host. These two parameters can be obtained since each host is equipped with a GPS device. When the timer expires, the host will wake up to see whether it is leaving the current grid. If it is leaving, the non-gateway host will send a unicast message to the gateway host to update the routing and host tables. The host must remain active until it finds another gateway. If it is not leaving, it will recalculate the dwell duration, set the timer, and then enter sleep mode again.

Two mobility situations should be addressed.

1. Hosts move into a new grid: Hosts will broadcast a $HELLO$ message when they move into a new grid. The gateway host in each grid will also broadcast its $HELLO$ message when it hears the $HELLO$ message. After the gateway's $HELLO$ message is received, the new incoming host will decide if it should replace the gateway. In this situation, only the host with a battery level that is higher than that of the original gateway can replace the original gateway. This rule prevents frequent replacement of gateways: such replacement is an overhead of our protocol. If a gateway must be replaced, then the new gateway will declare itself by sending a $HELLO$ message with the $gflag$ set. The original gateway, receiving this $HELLO$ message, will transmit the routing and host tables to the new gateway. If no gateway is replaced, then the new incoming hosts will enter sleep mode to conserve battery energy. If a new host does not receive any $HELLO$ message during a $HELLO$ period, the new host is in an empty grid and will declare itself as the gateway.

2. Hosts move out of a grid: The following considers the case of either a host's or a gateway's leaving one grid and entering another. A gateway must transfer its routing table to a new gateway before it leaves a grid. The gateway thus first sends a broadcast sequence to wake up all the hosts in the same grid. Only sleeping hosts in the same grid will wake up. After waiting for time, τ, the gateway will declare its departure by broadcasting a $RETIRE(grid, rtab)$ message, where *grid* represents the grid coordinate of the gateway, and $rtab$ represents the routing table. After receiving this $RETIRE$ message, all non-gateway hosts will store the routing table, and apply the gateway election algorithm to elect a new gateway. Then, the new gateway transmits a $HELLO$ message with the $gflag$ set to inform all the hosts. If a non-gateway host leaves a grid, it must notify the gateway about its departure by sending a unicast message to the gateway. The gateway will update its routing and host tables after receiving this unicast message.

In addition to host mobility, gateway replacement also takes place for load balance purpose. All hosts in the same grid should share the job of the gateway to extend the lifetime of a mobile host and the network. The load balance scheme presented here releases a gateway from its gateway job when its battery level changes, i.e., from upper to boundary, or from boundary to lower. The process by which a gateway quits its duty is the same as the process which is necessary when a gateway leaves a grid. A gateway will also quit its responsibility when it is going to exhaust its energy resource. That is, if a host is elected as the gateway with a lower battery level, it will act as the gateway until its battery is empty. Of course, the gateway will issue a broadcast sequence and a $RETIRE$ message before its battery runs out.

In case a gateway is down because of an accident and the $RETIRE$ message is not issued in time, the *no-gateway* event is occurred. Such a *no-gateway* event can be detected by a host when one of the following three situations happen: 1) an active host within the same grid has not received the gateway's $HELLO$ message longer than the regular interval, 2) a sleeping host awakes to transmit but does not get any response from the gateway, and 3) a host moving into the grid does not receive the gateway's returning $HELLO$ message after sending its $HELLO$ message. Once a *no-gateway* event is detected, the gateway election algorithm will be triggered.

3.3 Route Discovery and Data Delivery

In ECGRID, the routing table is established in a grid-by-grid manner, instead of in a host-by-host manner. Therefore, only the gateway is needed to maintain the routing table. Our ECGRID is an extension GRID (which is modified from AODV protocol [3]) by considering energy-conservation. The gateway is the only host in a grid that is responsible for the routing discovery procedure. Packets sent to a sleeping host are buffered at the gateway while the destination host is sleeping. Then, the gateway is responsible for waking up the destination host and forwarding these buffered data packets to it. If a sleeping host, S, must send data, it will wake up and send an acquire message $ACQ(gid, D)$ to inform the gateway, where gid represents its grid coordinate, and D represents the destination host. The gateway of S will respond with a $HELLO$ message and start the routing discovery procedure after receiving this ACQ message. Thus, host S can send data through the gateway to the destination. This handshaking is required since the gateway may be changed when a non-gateway host is sleeping.

When a source host, S, needs a route to a destination host, D, it will broadcast a route request, $RREQ(S, s\_seq, D, d\_seq, id, range)$ packet to request a route to D. The pair (S, id) can be used to detect duplicate $RREQ$ packets from the same source, S, avoiding endless flooding of the same request. The source sequence number, s_seq, represents the freshness of a reverse route from the destination to the source, and the destination sequence number, d_seq, indicates the freshness of the route from the source to the destination. The freshness information is used to determine whether a route is acceptable. The parameter, *range*, confines the area of search to where only the gateways within the area will participate the route searching procedure from S to D. The searching area limits the broadcast packets and thus alleviates the *broadcast storm problem* [13]. Several ways of confining the searching area have been presented in [2]. Routes may fail to exist in the searching area. In such a situation, another round of route searching should be initialized to search all areas for a route. Notably, a global search for a route is also needed when the source does not have location information concerning the destination.

When a gateway receives an $RREQ$ packet, the gateway will first check whether it is within the area defined by $range$. The gateway simply ignores this packet if the received packet is not within its range. Otherwise, the gateway checks the destination, D, from its routing table. If the destination, D, is in its routing table, then the gateway will rebroadcast the packet and set up a reverse pointer to the grid coordinate of the previous sending gateway. When D (or its gateway, if D is not a gateway,) receives this $RREQ$, it will unicast a reply packet $RREP(S, D, d\_seq)$ back to S through the reverse path. Notably, the reverse path is established when the $RREQ$ packet is broadcast. A gateway that receives the $RREP$ will add an entry to its routing table to specify that a route to D is available through the grid coordinate from which it received the $RREP$ packet. A route from S to D is properly established when this $RREP$ reaches S. When destination D is a non-gateway host, the gateway of D must wake D before forwarding data packets to it.

Fig. 2 shows an example to demonstrate how our protocol works. Suppose that the searching area is the smallest rectangle that can cover the grids of source, S, and destination, D. Initially, all hosts are active and each of them will broadcast a $HELLO$ message. After a $HELLO$ period, hosts S, A, B, C, D, E, F, and I will be selected as the gateway of grid (1,1), (1,2), (2,2), (2,1), (5,3), (3,2), (4,2), and (0,2), respectively, by applying the gateway election protocol. After the gateway hosts are elected, non-gateway hosts J, K, L, H, G and M can enter sleep mode to conserve energy. Suppose that host S wants to communicate with destination host, D; it will send an $RREQ$ packet which specifies that the searching area is the rectangle bounded by grids (1, 1), (1, 3), (5, 1), and (5, 3). When host B receives this $RREQ$ package for the first time, it will rebroadcast this packet, since host B is within the searching area. The reverse path which points to the grid (1, 1), in which S belongs, is recorded in host B. Similar actions (rebroadcasting $RREQ$ and recording a reverse path) will be performed at hosts E and F. The solid arrows in Fig. 2(a) show how the $RREQ$ packets move. The packet is finally reached at the destination, D, and a reverse path from D to S is also established. When destination D receives the $RREQ$, it will

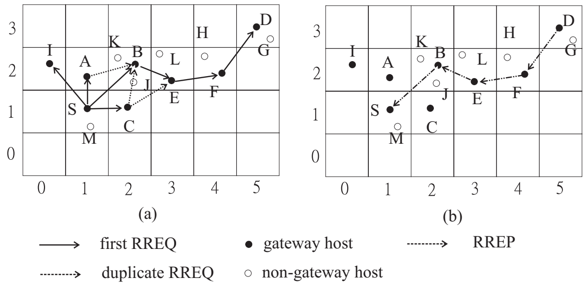


Figure 2. An example of routing discovery (a) propagation of $RREQ$ and (b) propagation of $RREP$ packets.

unicast an $RREP$ packet as a reply to S through the reverse path that was established by the $RREQ$ packets. Fig. 2 (b) shows the progress of $RREP$.

In the above example, destination D is a gateway. Data transmission through S-B-E-F-D can properly take place. Assuming that the destination is a non-gateway host, G, the route discovery process does not change and the path remains S-B-E-F-D. However, the data transmission path is S-B-E-F-D-G. The gateway, D, is responsible for waking G up and buffer data packets are sent to G before G is ready to receive.

3.4 Route Maintenance

This subsection considers the maintenance of a route when the source or destination leaves its original grid. The purpose of route maintenance is to keep the route available. Notably, the source or destination host becomes a non-gateway host when they move into a new grid. Suppose that the source host which is either a gateway host or a non-gateway host roams from grid g_1 to another grid, and that grid g_2 is the next grid along the route to the destination, as shown in Fig. 3. Hosts A, B, C, and E are the gateways of grids (1, 2), (2, 2), (2,1), and (3, 2), respectively. The following discusses four aspects of the route maintenance-protocol according to the roaming direction of the source host.

1. The source host moves into grid g_2, as shown in Fig. 3(a). The route can still function correctly. When S roams into grid g_2, it decides whether it will replace host B as the gateway. If replacement occurs, then S inherits the routing table from B. Thus, the route through B is maintained. If S does not replace B as the gateway, then $S's$ data will be transmitted with the help of its gateway, host B. The route can certainly work properly.

2. The source host S moves into a grid, g, that neighbors grid g_2, as shown in Fig. 3(b). In this case, if S becomes the gateway of g, then no change is needed. Otherwise, S becomes a non-gateway host and all data packets are forwarded by the gateway of the new grid g, A in grid (1, 2) or C in grid (2, 1). In this case, a new $RREQ$ is sent from A or C to B.

3. The source host moves into a grid, g, which does not neighbor grid g_2, and a gateway exists in grid g_1 as shown in Fig. 3(c). If S becomes the gateway of the new grid, g, then the source host changes its route entry for destination D to grid g_1. Then all data packets are forwarded to grid g_1. If S is a non-gateway host in the new grid, then data packets of S are forwarded by the gateway of the new grid, g. In such a case, a new $RREQ$ is sent to the gateway of grid g_1. In either case, the route is one hop longer than the original route.

4. The source host moves into a grid, which does not neighbor grid g_2, and a gateway does not exist in grid g_1, as depicted in Fig. 3(d). Therefore, no host is available to forward the source host's packets, and thus the route will be considered broken. The source host S initiates a new route discovery procedure to request a route to destination host D.

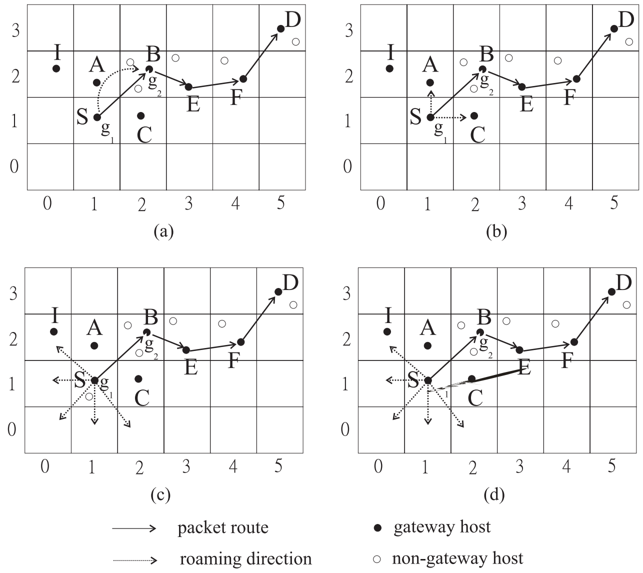
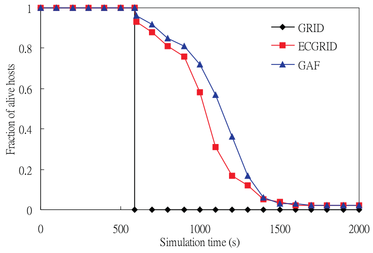


Figure 3. Route maintenance when the source host (gateway) roams off its current grid.

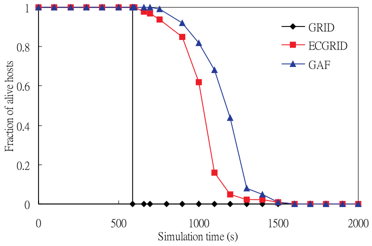
The rules for the movement of the destination host are similar to those for the movement of source host.

4 Simulation Results

The proposed ECGRID protocol is evaluated by an ns-2 simulator (CMU wireless and mobile extensions [14]). The simulation runs on a region of 1000×1000 square meters, a bandwidth of 2Mbps and a transmission range of 250 meters. Hosts move according to the random waypoint model, in which the hosts randomly choose a speed and move to a randomly chosen position. Then the hosts wait at the position for the pause time, before they start to move to the next randomly chosen location and speed. Two kinds of movement speeds are selected - one uniformly distributed between 0 and 1 m/s and one distributed between 0 and 10 m/s. The energy consumption model described in [10] is employed, which uses measurements taken by the Cabletron Roamabout 802.11 DS High Rate network interface card that operates at 2 Mbps. The power consump-

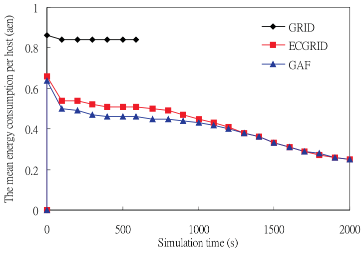


(a) roaming speed = 1 m/s

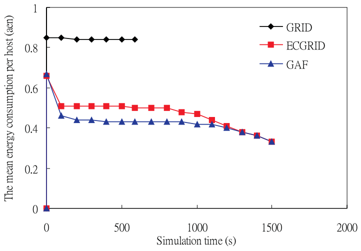


(b) roaming speed = 10 m/s

Figure 4. Fraction of alive hosts vs. the simulation time for GRID, GAF and ECGRID: (a) roaming speed = 1 m/s (b) roaming speed = 10 m/s. The number of hosts is 100 and the network traffic load is 10 pkts/s with constant mobility (pause time 0).



(a) roaming speed = 1 m/s



(b) roaming speed = 10 m/s

Figure 5. The mean energy consumption per host (aen) vs. the simulation time for GRID, GAF and ECGRID: (a) roaming speed = 1 m/s (b) roaming speed = 10 m/s. The number of hosts is 100 and the network traffic load is 10 pkts/s with constant mobility (pause time 0).

tion in transmit mode, receive mode, idle mode, and sleep mode is 1400mW, 1000mW, 830mW, and 130mW, respectively. The cost of receiving paging signals is ignored. The grid size is set to 100; therefore a gateway at the center of a grid can communicate with any gateway of its eight neighboring grids. Since each host is equipped with a GPS, we need to take into account the GPS power consumption. The GPS energy cost for GRID, ECGRID and GAF are all 0.033W[9].

Each source host sends a CBR (constant bit rate) flow with one or ten 512-byte packets per second. As mentioned in Section 1, if the destination is a host running GAF protocol, a lot of lost packets may be produced because the destination is sleeping. To keep the results comparable, we define two host models in our simulation in favor of GAF protocol.

- Model 1: Ten infinite energy hosts act as sources or destinations. These hosts do not run GAF protocol nor do they forward traffic. Another 100 hosts each with an initial energy 500 Joules, and which run GAF protocol, are used to evaluate energy consumption. This model applies only to GAF protocol because a source or destination host must be always active in GAF protocol.
- Model 2: Hosts with an initial energy of 500 Joules are used to evaluate energy consumption. All hosts run ECGRID or GRID protocol. The source and destination hosts are randomly chosen. The number of hosts is 50, 100, 150, or 200. This model is used by ECGRID and GRID.

Four observations can be made.

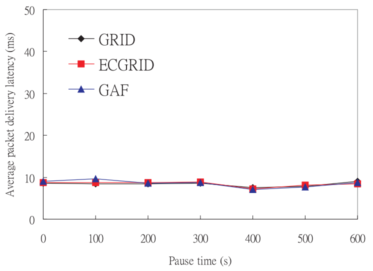
A) Effect of network lifetime:

In this experiment, the network lifetime of different protocols is observed. The network traffic load is 10 pkts/s with constant mobility (pause time is 0 second). In Fig. 4(a), the hosts' roaming speed is set to 1 m/s. The network that runs GRID, which is not energy-conserving, is down when the simulation time = 590 seconds. Both ECGRID and GAF prolong the network lifetime. Since each active host in ECGRID must periodically send a HELLO message to maintain the host table, GAF is more energy-conserving than ECGRID. For example, at speed = 1 m/s and simulation time = 800 seconds, 85% and 81% of hosts are alive for GAF and ECGRID, respectively. The increased power consumption results from the exchanging of the HELLO message. The HELLO message exchanged in ECGRID collects the hosts information in the same grid, thus guarantees successful data transmission, which cannot be achieved by GAF. The GAF assumes that the destination hosts are always in active mode when the sources send data to the destinations. We comment that the ECGRID is a practical and complete routing protocol although the fraction of live hosts is lower than that of GAF. Fig. 4(b) presents the same simulation with a roaming speed of 10 m/s. The result is similar to those in Fig. 4(a).

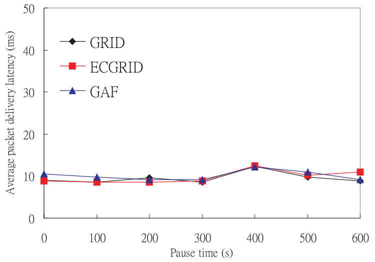
B) Effect of energy consumption:

In addition to the fraction of alive hosts, the mean energy consumption per host is also used for observing the effect of energy conserving. The mean energy consumption per host (aen) [9] is defined as,

$$aen = \frac{E_0 - Et}{n * t} \tag{2}$$

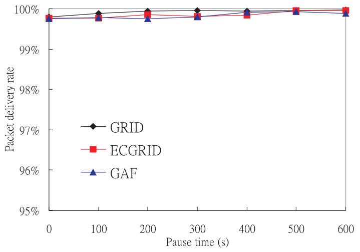


(a) roaming speed = 1 m/s

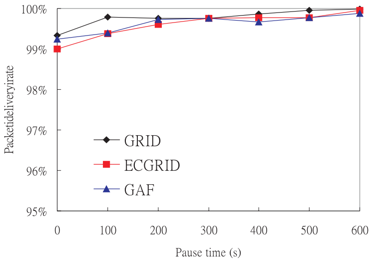


(b) roaming speed = 10 m/s

Figure 6. The packet delivery latency of GRID, GAF and ECGRID for various pause time: (a) roaming speed = 1 m/s (b) roaming speed = 10 m/s. The number of hosts is 100 and the network traffic load is 10 pkts/s.



(a) roaming speed = 1 m/s



(b) roaming speed = 10 m/s

Figure 7. The packet delivery rate of GRID, GAF and ECGRID for various pause time: (a) roaming speed = 1 m/s (b) roaming speed = 10 m/s. The number of hosts is 100 and the network traffic load is 10 pkts/s.

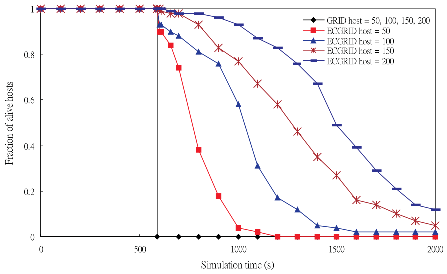
where E_0 represents the initial energy of the entire mobile hosts and E_t means the remaining energy of the entire hosts after time t. The number of hosts in the network is n. The network traffic load is set to 10 pkts/s with constant mobility (pause time 0 second) and the aen are calculated for different protocols. The results for a roaming speed of 1 m/s and 10 m/s are shown in Fig. 5(a) and (b), respectively. These two Figs. have the similar curves that the aen for GRID is, when the simulation time is prior to 590 seconds, about 33% and 38% higher than that of ECGRID and GAF, respectively. Therefore, we conclude that an energy-aware routing protocol, such as ECGRID or GAF, can consume less energy than GRID at different roaming speeds.

C) Effect of packet delivery:

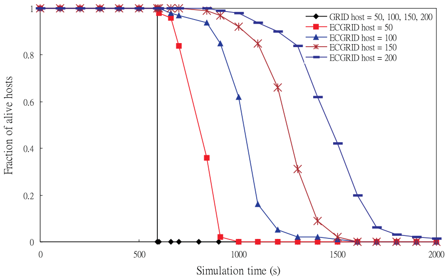
Next, the *packet delivery rate* and *average packet delivery latency* are observed at different pause times. The packet delivery rate is defined as the number of data packets actually received by the destination, divided by the number of packets issued by the corresponding source host. The average packet delivery latency is defined as the average time elapsed between packet transmission and reception. The packet delivery qualities, packet delivery rate and end-to-end delay (latency), are compared for these three protocols with simulation time = 590 seconds, since the network hosts that run GRID exhaust all their energy at simulation time = 590 seconds. The results shown in Fig. 7 and Fig. 6 are based on a network traffic load of 10 pkts/s at roaming speeds of 1 m/s and 10 m/s. Fig. 7 reveals that the packet delivery rate exceeds 99% for all three protocols at a speed of 1m/s or 10m/s. Note that the destinations are always active in GAF protocol, and thus the delivery rate of GAF will be reduced if the destinations are not always active as ECGRID does. Fig. 6 shows that all three protocols have a similar average packet delivery latency, between 7.1 ms and 10.7 ms at a speed of 1 m/s and between 8.5 ms and 12.5 ms at a speed of 10 m/s. This experiment shows that our power-conserving protocol ECGRID can achieve its target without reducing the quality of delivered packets.

D) Effect of host density:

In this experiment, the host density is varied to elucidate the relation between network lifetime and host density. The host number is set to 50, 100, 150 and 200. The network traffic load is 10 pkts/s with constant mobility (pause time is 0 second). Fig. 8(a) and (b) show the results at a roaming speed of 1 m/s and 10 m/s, respectively. The network lifetime in GRID is observed to be the same for various host densities, because the network does not conserve energy. The network lifetime of our protocol increases with the host density, because only one host (gateway) in a grid is active. As the grid contains each host can take turns to act as the gateway. Comparing Fig. 8(a) and (b) shows that a higher roaming speed corresponds to better load balance between hosts. For example, with host number = 200, hosts start to run out of energy at simulation time = 660 seconds for a roaming speed of 1 m/s (Fig. 8(a)) and at simulation time = 910 seconds for a roaming speed of 10 m/s (Fig. 8(b)). The network lifetime is longer at a lower roaming speed, because a higher roaming speed of each host leads to frequent gateway selection. The gateway selection will be restarted (if a gateway host leaves its original grid) or a unicast message will be sent (if a non-gateway host leaves its original grid). The frequent gateway selection consumes much energy.



(a) roaming speed = 1 m/s



(b) roaming speed = 10 m/s

Figure 8. The fraction of alive hosts affected by host density for GRID and ECGRID: (a) roaming speed = 1 m/s (b) roaming speed = 10 m/s. The host number is varied from 50 to 100, 150 and 200. The network traffic load is 10 pkts/s with constant mobility (pause time = 0).

5 Conclusions

The issue of energy conservation is critical in a limited energy resource MANET. This study proposes a novel energy-aware routing protocol, ECGRID, for mobile ad hoc networks. ECGRID extends the GRID protocol to account for energy constraints. One is elected as a gateway in each gridto handle route discovery and packet delivery. Energy is conserved by turning the non-gateway hosts' transceivers off when the hosts are idle. A gateway host can awaken sleeping hosts through Radio Frequency tags technology. Accordingly, sleeping hosts need not wake up periodically. A load balance of the mobile host's battery energy scheme is applied to prolong the lifetime of all mobile hosts. Also, ECGRID eliminates the limitation that destination hosts must always be active (as is assumed for earlier protocols, such as GAF). Simulation results demonstrate that ECGRID can not only prolong the lifetime of the entire network but also maintain good packet delivery ratio. A host runs ECGRID consumes less energy than a host runs GRID does. Additionally, the lifetime is extended in proportion to the host density in the whole network.

References

[1] J. Broch, D. B. Johnson, and D. A. Maltz, "The dynamic source routing protocol for mobile ad hoc networks (Internet draft)," Feb. 2002.

[2] W.-H. Liao, Y.-C. Tseng, and J.-P. Sheu, "GRID: a fully location-aware routing protocol for mobile ad hoc networks," *Telecommunication Systems*, vol. 18, no. 1, pp. 37–60, Sep. 2001.

[3] C. Perkins and E. M. Royer, "Ad hoc on demand distance vector (AODV) routing (Internet draft)," Jan. 2002.

[4] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector (DSDV) routing for mobile computers," in *ACM SIGCOMM Symposium on Communications, Architectures and Protocols*, Sep. 1994, pp. 234–244.

[5] P. J. Wan, G. Galinescu, X. Y. Li, and O. Frieder, "Minimum-energy broadcast routing in static ad hoc wireless networks," in *Proc. of IEEE INFOCOM, vol.2*, 2001, pp. 1162–1171.

[6] V. Rodoplu and T. H. Meng, "Minimum energy mobile wireless networks," in *Proc. of the 1998 IEEE International Conference on Communications (ICC'98), vol.3*, June 1998, pp. 1633–1639.

[7] R. Wattenhofer, L. Li, P. Bahl, and Y.-M. Wang, "Distributed topology control for power efficient operation in multihop wireless ad hoc networks," in *Proc. of IEEE INFOCOM, vol.3*, 2001, pp. 1388–1937.

[8] S.-L. Wu, Y.-C. Tseng, and J.-P. Sheu, "Intelligent medium access for mobile ad hoc networks with busy tones and power control," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 9, pp. 1647–1657, Sep. 2000.

[9] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed energy conservation for ad hoc routing"," in *Proc. of the 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'01)*, 2001, pp. 70–84.

[10] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks," in *Proc. of the 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'01)*, 2001, pp. 85–96.

[11] L. M. Feeney and M. Nillsson, "Investigating the energy consumption of a wireless network interface in an ad hoc networking environment," in *Proc. of IEEE INFOCOM, vol.3*, 2001, pp. 1548–1557.

[12] C. F. Chiasserini and R. R. Rao, "Combining paging with dynamic power management," in *Proc. of IEEE INFOCOM, vol.2*, 2001, pp. 996–1004.

[13] Sze-Yao Ni, Yu-Chee Tseng, Yuh-Shyan Chen, and Jang-Ping Sheu, "The broadcast storm problem in a mobile ad hoc network," in *Proc. of the 5th ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99)*, Aug. 1999.

[14] "The cmu monarch project," http://www.monarch.cs.cmu.edu/.