

A Hybrid Power-Saving Protocol by Dual-Channel and Dual-Transmission-Range Clustering for IEEE 802.11-Based MANETs

Jehn-Ruey Jiang^{*}, *Member, IEEE*

Chau-Yuan Yang[†]

Ting-Yao Chiou[‡]

Shing-Tsaan Huang[§], *Fellow, IEEE*

Department of Computer Science and Information Engineering

National Central University, Jhongli city, 320, Taiwan, R.O.C

Email : {^{}jrjiang, [§]sthuang}@csie.ncu.edu.tw, [†]des@xpl.csie.ncu.edu.tw, [‡]tingyao@acnlab.csie.ncu.edu.tw*

Abstract—There are two types of MAC layer power-saving (PS) protocols for IEEE 802.11-based MANETs: synchronous and asynchronous ones. In this paper, we propose a hybrid PS protocol to take advantages of both types of protocols. The protocol utilizes the concept of dual-channel and dual-transmission-range clustering. It divides all the hosts into clusters. Each cluster has a head and all the heads are organized as a virtual backbone to help route data. The protocol also utilizes the cluster head dismissal mechanism to avoid the ever-increasing of cluster heads and to adapt to topology changing. Simulation results demonstrate that the proposed protocol is more power-efficient and more scalable than related protocols.

Index Terms—IEEE 802.11, power saving, mobile ad hoc network (MANET), virtual backbone

I. INTRODUCTION

The mobile ad hoc network (MANET) has attracted a lot of attention recently. A MANET consists of a set of mobile hosts, and does not have the support of any base station. Hosts have unpredictable mobility and can communicate with each other by sending messages either over a direct wireless link, or over a sequence of wireless links including one or more intermediate hosts. Applications of MANETs include battlefield communications, disaster rescue operations, outdoor activities, and so on.

Power saving is a critical issue for portable devices supported by batteries. This is because battery power is a limited resource, and battery technology is not likely to progress as fast as computing and communication technologies do. How to save the energy consumption in a MANET, in which devices are all supported by batteries, has been intensively studied recently (e.g., power control is studied in [5, 6, 12, 21, 23], power-aware routing in [3, 14, 15, 19], and low-power mode management in [1, 2, 4, 7, 8, 10, 16, 17, 18, 22, 24]).

This paper investigates the low-power mode management problem for IEEE 802.11-based MANETs. There are two types of power-saving (PS) protocols for such MANETs: synchronous and asynchronous ones. IEEE 802.11 [9] proposes its own protocol for single hop (full connected) MANETs based on periodical transmission of beacons. The

protocol requires accurate clock synchronization and is classified as the synchronous protocol. Such a protocol is only suitable for single-hop environments since clock synchronization for multi-hop networks is costly and even impossible [20]. On the other hand, the papers [7] and [20] propose asynchronous PS protocols which need not to synchronize host clocks. Among the protocols, the quorum-based asynchronous power-saving (QAPS) protocol proposed in [7] usually has the lowest *active ratio*, the percentage of time for a host to turn on its radio. Although the asynchronous protocols do not need to synchronize all hosts, they usually consume more energy than the synchronous one.

In this paper, we propose a hybrid power saving protocol to combine the advantages of synchronous and asynchronous ones. According to the proposed protocol, all the hosts are divided into clusters. Each cluster has one cluster head with others being cluster members. The members are one-hop neighbors of the head and are synchronous with it. The IEEE 802.11 synchronous PS protocol is operated within an individual cluster. And the QAPS protocol is operated among cluster heads. All the cluster heads are organized as a virtual backbone to help route data. We perform simulations and compare the protocol with the related protocols. We find that the proposed protocol is more power-efficient and more scalable.

The contributions of this paper are four-fold. First, it utilizes the clustering concept to design a hybrid PS protocol taking advantages of synchronous and asynchronous PS protocols. Second, it adopts dual-transmission-range concept to eliminate the need of selecting gateway for relaying inter-cluster messages. Third, it adopts dual-channel concept to allow simultaneous transmissions of inter-cluster and intra-cluster messages, which increases throughput and facilitates cluster maintenance. And fourth, it uses the cluster head dismissal mechanism to void the ever-increasing of cluster heads, and to make the protocol adaptive to topology changing.

The rest of this paper is organized as follows. Preliminaries are given in Section 2. Section 3 introduces the proposed protocol. Simulation results are presented in Section 4. Conclusions are drawn in Section 5.

II. PRELIMINARIES

IEEE 802.11 supports two power modes: active and power-saving (PS). Under the PS mode, a host can reduce its radio activity by only monitoring some periodical signals (such as beacons) in the network. Tuning a host to the PS mode can save a lot of energy. For example, for an ORiNOCO IEEE 802.11b PC GoldCard, the power consumption is 1400 mW, 950 mW and 805 mW when it remains active to transmit, receive, and monitor data packets, respectively. When the card switches to PS mode, the power consumption can be reduced to 60 mW.

Under the ad hoc mode, IEEE 802.11 PS protocol divides the time axis into equal-length beacon intervals, each of which starts with an ATIM (Ad hoc Traffic Indication Map) window. The ATIM window is relatively small compared to the beacon interval. PS hosts must remain active during the ATIM window so as to be notified by those intending senders, and may go to PS mode in the rest of the beacon interval if no one intends to send packets to it (refer to Fig. 1).

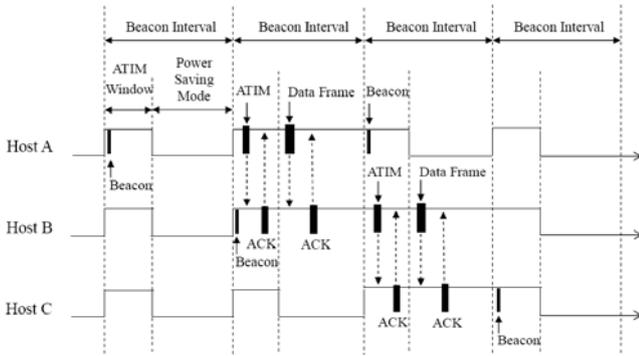


Fig. 1. Transmission scenarios for PS hosts in a single-hop IEEE 802.11 MANET.

The power saving protocol of IEEE 802.11 only considers single-hop MANETs. It is classified as a *synchronous* protocol. On the other hand, the papers [7] and [20] propose several *asynchronous* power saving protocols, which are suitable for multi-hop MANETs. Among them, the quorum-based asynchronous power saving (QAPS) protocol is most promising because it sends the least number of beacons and usually has the lowest active ratio. According to the QAPS protocol, the time axis is divided evenly into beacon intervals, which are classified into two types (refer to Fig. 2): (1) Quorum interval: It starts with a beacon window followed by a MTIM window. After the MTIM window, the host remains active (in monitor mode) for the rest of the beacon interval. (2) Non-quorum interval: It starts with a MTIM window. After the MTIM window, the host may go to the PS mode if it has no packets to send or to receive.

Similar to IEEE 802.11, the beacon window is for hosts to compete to send their beacons. The MTIM window is similar to the ATIM window — it is for a host with buffered packets to compete to send notifications to intended receivers in the PS mode to wake them up. It is named so to reflect that it is used for multi-hop ad hoc networks.

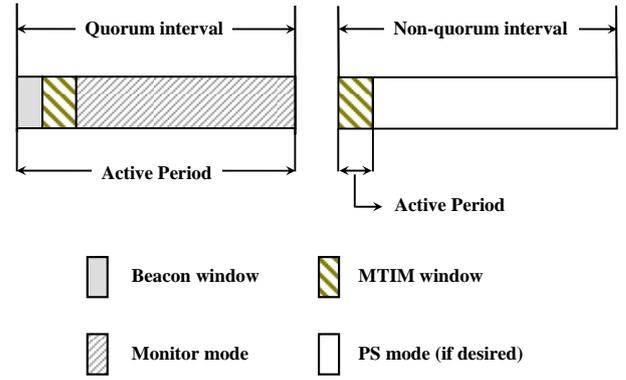


Fig. 2. Structures of quorum intervals and non-quorum intervals.

In [20], it is proposed to group every n consecutive intervals into a *round*, and to use the grid quorum system for a host to select quorum intervals. The paper [7] further shows that the finite projective plan (FPP) quorum system, the grid quorum system, the torus quorum system, and the cyclic quorum system can be used for the quorum interval selection. A quorum system is a collection of subsets of a universal set, where each subset is called a *quorum*. Quorums should be *minimal* (i.e., every quorum is not a super set of another quorum) and satisfy the *intersection property* that any pair of quorums should have non-empty intersection. For example, $Q = \{ \{0,1,2\}, \{1,5,6\}, \{2,3,6\}, \{0,4,6\}, \{1,3,4\}, \{2,4,5\}, \{0,3,5\} \}$ is a finite projective plan quorum system under the universal set $\{0, \dots, 6\}$ (such a universal set corresponds to the situation where 7 consecutive intervals are grouped into a round). By the intersection property and the specific design of beacon intervals as depicted in Fig. 2, it is shown in [7] and [20] that no matter how asynchronous hosts' clocks are, two neighboring hosts can hear each other's beacon at least once in every round. By embedding clock information in beacon frames, a host can figure out others' wake-up time by evaluating the clock difference so that it can initiate a data transmission at the proper time when the receiver turns on its radio.

As we have mentioned, among the asynchronous PS protocols, the quorum-based one — the QAPS protocol — usually has the lowest active ratio. It also has the advantage of being applicable to multi-hop MANETs. Below, we show two observations about the QAPS protocol:

- (1) Its active ratio is still much higher than the synchronous PS protocol.
- (2) A high host density has negative impacts on the QAPS protocol. It causes high broadcast traffic and short network life time.

III. THE PROPOSED PROTOCOL

In this section, we propose a protocol to respond to the observations about the QAPS protocol. The protocol is intended to be power-efficient and to be suitable for dense networks. Below, we first show the overview of the proposed protocol.

A. Overview

The basic idea of the proposed protocol is to integrate the

synchronous and the asynchronous power saving protocols in IEEE 802.11-based MANETs for saving more energy and accommodating more hosts. Between the two types of the PS protocols, the synchronous PS protocol has lower active ratio but is not suitable for multi-hop MANETs, while the asynchronous protocol is a contrary. So we try to combine the advantages of both PS protocols. We propose to divide all the hosts into clusters, in each of which one host is selected as the *head* with others being the *members*. The synchronous PS protocol is operated within an individual cluster and the QAPS protocol is operated among cluster heads. We demand the clustering mechanism not to rely on location information so that it can be applied to MANETs of hosts without positioning systems. We also demand it to operate only on the basis of neighborhood information in order that it can adapt to network topology changes as quickly as possible.

Each host can switch between two non-interfering communication channels: A and B, and can switch between two different communication ranges: R_A and R_B . The cluster head uses channel A with transmission range R_A to communicate with other cluster heads, and uses channel B with transmission range R_B to communicate with its cluster members. A host will go through a lot of states as depicted in Fig. 3. We will elaborate the details of the state transition throughout this section.

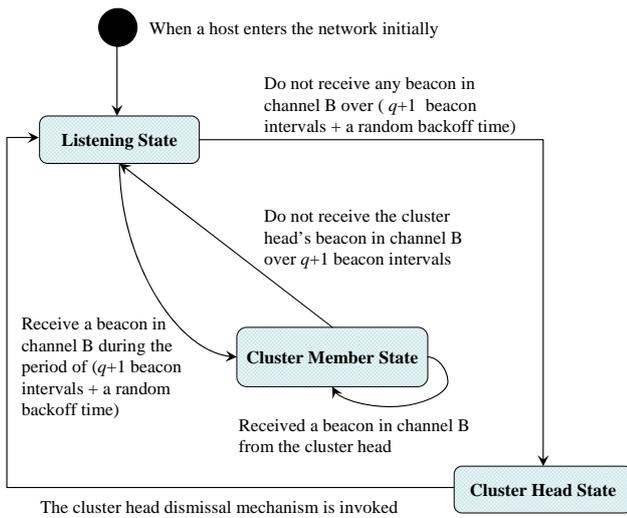


Fig. 3. The state transition diagram for hosts.

B. Clustering Mechanism

Similar to the IEEE 802.11 PS protocol, the time axis is divided into equal-length beacon intervals. Cluster heads send beacons in channel B with transmission range R_B in some specific beacon intervals. When a host powers on, it enters the *listening state* and keeps monitoring beacons in channel B for $(q+1)$ beacon intervals + a random backoff time, where q is the number of quorum intervals in a round and the backoff time takes 0 to 15 time slots (each slot time occupies 20 micro seconds). The reason why a host must keep monitoring beacons over $q+1$ beacon intervals will be explained later. If the host hears a beacon from a host h , it enters the *cluster member state*, joins h 's cluster as a cluster member, and synchronizes its clock with h . Otherwise it sets itself a new cluster head; it enters the *cluster head state*

and starts to send beacons. Note that the random backoff time is used to reduce the possibility that neighboring hosts claim themselves to be heads simultaneously. This is for the purpose of keeping the number of cluster heads as small as possible.

A host as h 's member will keep monitoring h 's beacons every beacon interval to judge whether h still exists or not (h may move away or power off). If the cluster member does not receive any beacon from h over $q+1$ beacon intervals, it assumes h is gone or itself out of the transmission range R_B of h . The cluster member then enters the listening state **again** for the purpose of joining another cluster or becoming a new cluster head. Accordingly, all the hosts will be divided into clusters; some of them are cluster heads and others are cluster members.

Recall that a host monitors beacons sent in channel B with transmission range R_B for deciding whether to join a cluster. Also recall that a cluster head uses channel A with transmission range R_A to communicate with other heads. We can see that the choices of R_A and R_B affect the connectivity of clusters and the number of clusters.

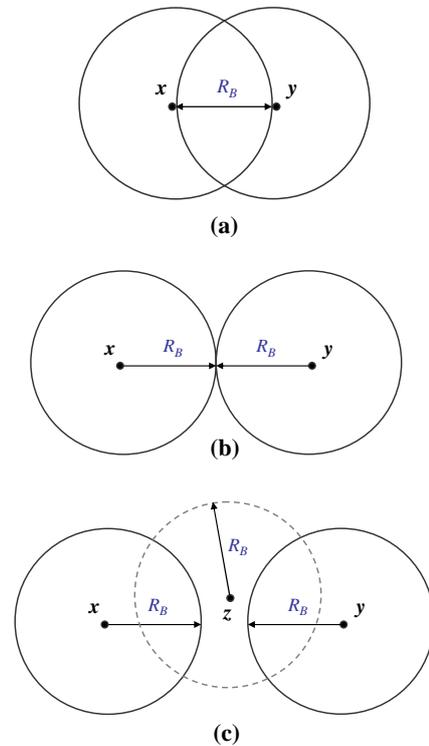


Fig. 4. The relationship between two cluster heads x and y .

Below, we discuss the relationship between R_A , R_B and the connectivity. In Fig. 4 (a) and (b), we show two extreme cases of the relation between a cluster head x and its neighboring cluster head y under the assumption that the host density goes to infinity. The minimum distance between x and y is R_B (refer to Fig. 4 (a)) because no host within the transmission range R_B of x can be a cluster head according to the proposed clustering mechanism. And the maximal distance is $2R_B$ (refer to Fig. 4 (b)). This is because any hosts outside the transmission range R_B of x can be a cluster head and y at distance of $2R_B$ from x is the farthest host to be the neighboring cluster head. If y goes farther, there will be another cluster head z between x and y (refer to

Fig. 4 (c)) and y is not the neighboring cluster head of x . If we restrict $R_B \leq \frac{1}{2} R_A$ (i.e. $R_A \geq 2R_B$), then it is guaranteed that a cluster head can always communicate with its neighboring cluster heads.

It is easy to check that the smaller R_B is, the more cluster heads are. Therefore we can set $R_B = \frac{1}{2} R_A$ to maintain network connectivity while keeping cluster heads as few as possible. However, when the host density is not infinite, connectivity may be violated (i.e. a cluster head may not communicate with any other cluster heads). One solution to the problem is to decrease R_B to reduce the probability of violating connectivity. But this will lead to more cluster heads, which in turn will cause more energy consumption. Since the proposed protocol is designed for high host-density networks, we may well still set $R_B = \frac{1}{2} R_A$ to trade the risk of violating connectivity for energy saving. Fortunately, as we simulate the proposed protocol for networks of $1000\text{m} \times 1000\text{m}$ area with 100, 200, \dots , 500 hosts, the network are connected almost all the time.

C. Cluster Head Dismissal Mechanism

Since the hosts move at random, cluster heads may gather together from time to time. When two or more cluster heads get too close, we can dismiss some of them from the duty of being heads. Each cluster head is assigned a totally ordered priority, which is a triplet of (the duration of being the cluster head, the negative of the residual energy, the node ID). The triplet is embedded in the beacon. It defines a lexical graphical order for hosts to be heads. When a cluster head hears beacons from another head, it would estimate the distance between itself and the head by the received signal strength. If the distance is smaller than a threshold, it will check its priority and determine whether to continue serving as a cluster head. According to the priority, a host x has more chance to be dismissed from being a head than a host y if x has been a cluster head longer than y , or if x has less residual energy than y (in case x and y have been heads for the same period of time), or if x has larger ID than y (in case x and y have been heads for the same period of time and they have the same residual energy). The distance threshold is adjustable; a larger threshold results in a smaller number of cluster heads but makes higher the variation of cluster heads, which is baneful to routing. We assume that the threshold is $1/10$ of the normal transmission range R_A in the following context.

D. Structures of Beacon Intervals

The structures of beacon intervals for the cluster head and the cluster member are different (refer to Fig. 5). The cluster head runs the QAPS protocol proposed in [7], and its beacon intervals are classified into the following two types:

1) *Quorum interval*: In the quorum interval, the cluster head uses channel A with transmission range R_A to communicate with other cluster heads. The quorum interval starts with a beacon window followed by a MTIM window. After the MTIM window, the cluster head remains active (in monitor mode) in channel A for the remaining of the beacon interval.

2) *Non-quorum interval*: Non-quorum interval starts with a beacon window followed by a MTIM window in channel A. Then, a beacon window (denoted by B') and a MTIM window (denoted by M') in channel B follow. After those windows, the cluster head may go to the PS mode if it has no packets to send or to receive. It is noted that during the beacon window and the MTIM window in channel B, the cluster head adjusts its transmission range to be R_B . The data exchange occurs after the second MTIM window. If the data exchange is between (or among) cluster heads, then channel A and transmission range R_A will be used. On the other hand, if the data exchange is for the cluster head and its members, channel B and transmission range R_B will be used.

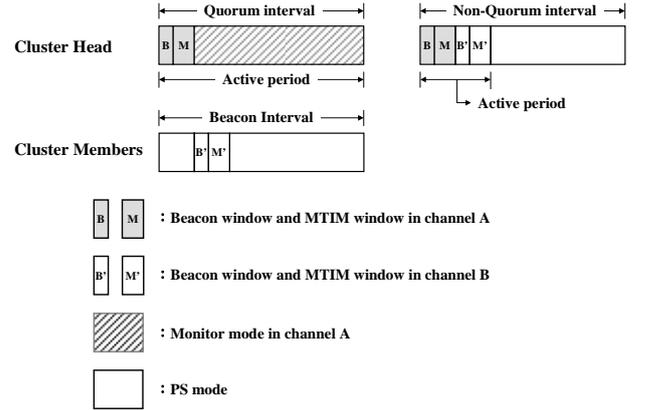


Fig. 5. The structures of beacon intervals for cluster heads and cluster members.

As to the cluster member, there is only one type of beacon interval for it. The beacon interval starts with a beacon window followed by a MTIM window in channel B. After the MTIM window, the cluster member may go to the PS mode if it has no packets to send or to receive. The cluster member uses channel B with transmission range R_B to communicate with the cluster head.

In summary, the cluster head usually operates in channel A with transmission range R_A ; it operates in channel B with transmission range R_B only during the second beacon window (B'), the second MTIM window (M'), and the data exchange period with cluster members in the non-quorum interval. On the other hand, the cluster member always operates in channel B with transmission range R_B . (However, as we will show later, an exception may be demanded by routing protocols.)

E. Synchronous Power-Saving within a Cluster

By the proposed protocol, each cluster member is a one-hop neighbor of the associated cluster head. So, we can run the synchronous PS protocol within an individual cluster. The protocol restricts that cluster members can only communicate with their associated cluster head in channel B with transmission range R_B . After joining a cluster, a cluster member copies the clock information of the cluster head. Therefore, the cluster member is synchronous with the cluster head.

As we have mentioned, the cluster head sends a beacon during the beacon window (B') in channel B with transmission range R_B in the non-quorum interval. Since the

cluster member is synchronous with the cluster head. It can thus wake up at proper time and uses channel B to monitor the beacons sent from the cluster head. The cluster member maintains a counter recording the number of times of not hearing a beacon in the beacon window. The counter is reset to zero when the cluster member hears a beacon. Once the counter is greater than a threshold $q+1$ (q is the number of quorum intervals in a round), the cluster member assumes that the head is absent. It then enters the listening state for the purpose of joining a new cluster or becoming a new cluster head.

The reason why we choose the threshold $q+1$ is explained below. A cluster head only sends a beacon in channel B during the second beacon window (B') in non-quorum intervals. Thus, in the worst case, there can be up to q quorum intervals preceding a non-quorum interval, which causes a cluster head **unable** to send beacons in channel B over consecutive q beacon intervals. So, not hearing a beacon sent in channel B over consecutive $q+1$ beacon intervals suffices for detecting the absence of the cluster head. This also accounts for the reason why the protocol demands a host to keep monitoring beacons in channel B over ($q+1$ beacon intervals + a random backoff time) when it enters the system initially and when it is in the listening state.

F. Asynchronous Power-Saving among Clusters

By the proposed protocol, each cluster head operates the QAPS protocol to communicate with other cluster heads. Each cluster head will send a beacon in channel A with transmission range R_A in the quorum interval. After this, it can send a MTIM message to its intended receiver in the MTIM window. A data exchange in channel A with transmission range R_A will then proceed after the MTIM window. A cluster head will keep in monitor mode in channel A for the remaining of the quorum interval if having no data to send or to receive. In the non-quorum interval, the cluster head monitors beacons in the first beacon window, and sends/receives MTIM messages in the first MTIM window in channel A with transmission range R_A . Since the cluster head obeys the QAPS protocol [7], it can discover any newly coming neighboring hosts within a round and can thus communicate with all neighboring hosts properly.

G. Virtual Backbone and Routing

We can treat the set of cluster heads as a virtual backbone through which the data are routed. Below, we adapt the well-know AODV (Ad hoc On-Demand Distance Vector) protocol [11] as the routing protocol to illustrate how the virtual backbone helps route data. AODV uses several types of messages, such as the route-request and the route-reply messages, for a source host to find a path to the destination host. The source host can be a cluster head or a cluster member. For the case that the source host is a cluster member, the source host at first contends to send an MTIM message to its cluster head in the MTIM window in channel B with transmission range R_B in non-quorum interval. If the cluster head successfully receives the MTIM message, it will respond with an ACK message. Afterwards, the source host can send the route-request message to the

cluster head. After receiving the route request message, the cluster head will broadcast the message to its local cluster members in channel B with transmission range R_B in the next non-quorum interval. If the destination host is in the local cluster, it will send a route-reply message to the cluster head immediately. Otherwise, if no immediate route-reply message is received, the cluster head will rebroadcast the route-request message to the neighboring cluster heads in channel A with transmission range R_A in the coming quorum intervals. When a cluster head receives the route-request message, it will locally broadcast the message to its cluster members and rebroadcast the message to all neighboring cluster heads in case that no route-reply from the **members** is received. For the case that the source host is a cluster head, the source host follows the same procedure as just mentioned except that it at first broadcasts the route-request message to its own cluster members in channel B with transmission range R_B in the non-quorum interval, and then rebroadcasts the message to all its neighboring cluster heads in channel A with transmission range R_A if no route-reply message is received. It is noted that only cluster heads rebroadcast the route-request message. This reduces the number of rebroadcasts dramatically, saves much energy, and avoids many collisions.

The route-request message rebroadcast will continue until the message reaches the destination host or a maximum hop count is encountered. When the message reaches the destination host, a route reply message will be issued, and tracked back to the source host according to the reverse path of receiving the route-request message. Afterwards, data can be sent out according to the path indicated by the route-reply message. To accelerate the data transmission, a host is restricted to remain active for receiving/relaying data when it issues, forwards or receives the route-reply message. All the hosts (either heads or members) participating the data transmission now switch to channel A with transmission range R_A for communicating with each other by RTS/CTS mechanism. We demand a cluster member to also operate with transmission range R_A so that the RTC/CTS mechanism can work properly. This is the exception demanded by the routing protocol for the cluster member to operate in channel A with transmission range R_A . Certainly, this can accelerate data transmission.

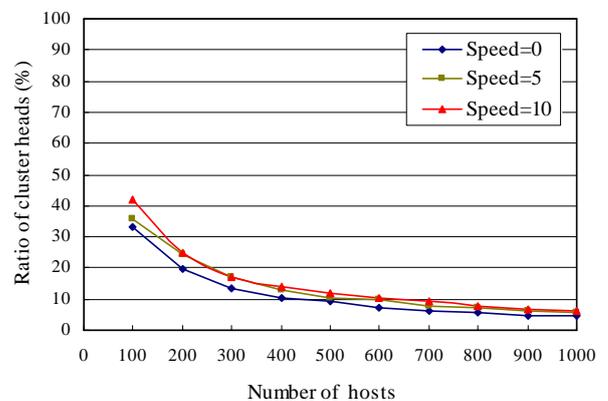


Fig. 6. The ratio of cluster heads for different speeds and host densities. (number of hosts = 100 ~ 1000, mobility = 0 ~ 10m/sec)

IV. SIMULATION RESULTS

In this section, we compare the proposed hybrid PS (HPS) protocol with the quorum-based asynchronous PS (QAPS) protocol [7] through a simulator written in C. The HPS protocol combines the synchronous PS protocol [9] and the QAPS protocol using torus(4×8) quorum system [7]. And the QAPS protocol for comparison is the one using torus(4×8) quorum system [7].

An area of size 1000m × 1000m is simulated. Each host has a transmission rate of 2M bits/sec, a transmission radius of 250 meters, and initial battery energy of 100 Joules. The MAC part basically follows the IEEE 802.11 standard [9], except the power management part. Routes with random sources/ destinations are generated, and the adapted AODV (ad-hoc on-demand distance vector) routing protocol is adopted.

Three parameters are tunable in our simulations:

- **Mobility:** Host mobility follows the random way-point model, with pause time of 20 seconds. When moving, a host's speed can range in 0 ~ 10 meters/sec.
- **Traffic load:** Routes are generated by a Poisson distribution with rates between 1 ~ 5 routes/sec. For each route, 20 packets, each of size 1K bytes, are sent.
- **Number of hosts:** The total number of mobile hosts in the MANET is 100 ~ 1000 hosts.

Three performance metrics are measured in the simulations:

- **The ratio of cluster heads.**
- **Survival ratio:** The number of surviving hosts (with non-zero energy) over the total number of hosts.
- **Throughput:** The average number of MAC-layer data packets successfully received in the network per second.

A host can go to the PS mode when it does not serve as a source, a destination, or a relay host of any route. A broadcast message (such as the route-request message) may need to be sent multiple times if the sending host finds that some of its neighbors are in the PS mode [7]. This is necessary because these PS hosts may wake up at different times and we need multiple transmissions to cover all of them. However, once a route is established (via the notification of a route reply message), all hosts in the route have to tune to the active mode.

Below, we show how mobility and host density affect the performance of the proposed HPS protocol. Fig. 6 shows the ratio of the number of cluster heads over the total number of hosts for different speeds and different host densities. From this figure we know that when the speed increases, the ratio of cluster heads increases slightly. However, the ratio of cluster heads is decreasing when host density increases. The ratio of cluster heads affects the consumption of power since higher cluster head ratio indicates more power consumption. Thus, our protocol is expected to have better performance in high host density environment. Fig. 7 (a) and Fig. 7 (b) illustrate the distribution of cluster heads. We can see that cluster heads

are evenly distributed over the whole area.

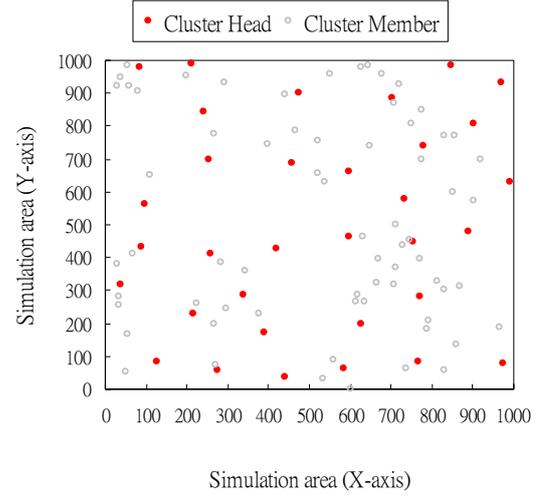


Fig. 7 (a). Distribution of cluster heads. (100 hosts in a 1000m x 1000m area)

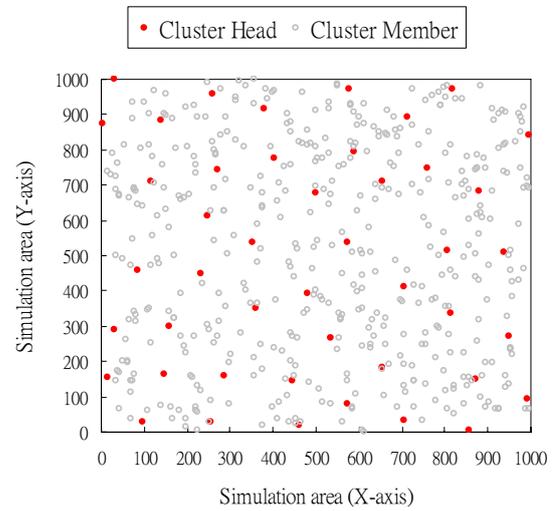


Fig. 7 (b). Distribution of cluster heads. (500 hosts in a 1000m x 1000m area)

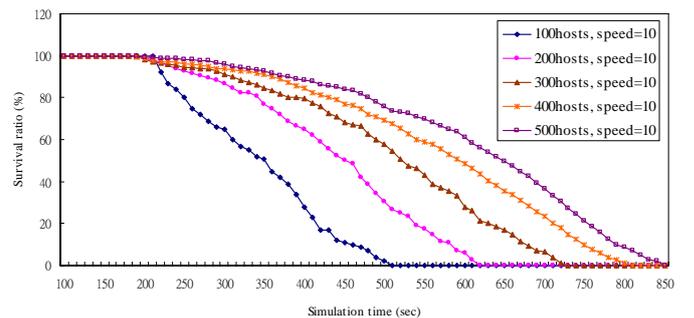


Fig. 8. Survival ratios of the proposed protocol for different numbers of hosts. (number of hosts = 100 ~ 500, mobility = 10m/sec)

Fig. 8 shows the host survival ratios of the proposed protocol for different numbers of hosts. We have observed that a larger number of hosts usually leads to a higher survival ratio. And Fig. 9 shows the survival ratios for different host mobilities. We have observed that a lower degree of mobility usually leads to a higher survival ratio. As shown in Fig. 10, however, our protocol outperforms the QAPS protocol (using torus(4×8) quorum system) in terms of the survival ratio. In these two figures, the curves of the proposed protocol are stair-wise. The stair-wise shapes are

caused by the simultaneous death (running out of energy) of some cluster heads that are elected as the heads almost at the same time. In the proposed protocol, cluster heads have much more loads than cluster members, and thus run out of energy soon. When the MANET bootstraps, some of the hosts are elected as cluster heads almost at the same time. And afterwards, the simultaneous death of cluster heads proceeds repeatedly. And this is the cause of the stair-wise survival ratio curves.

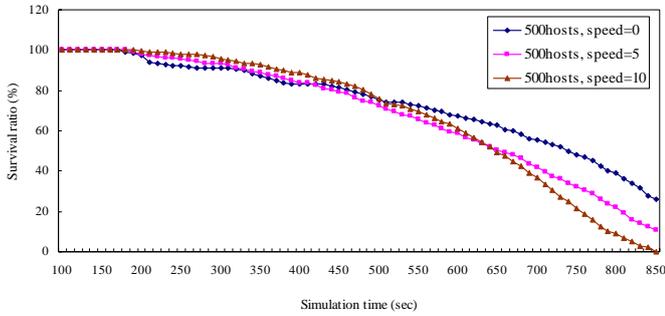


Fig. 9. Survival ratios of the proposed protocol for different host mobility. (number of hosts =500, mobility = 0 ~ 10m/sec)

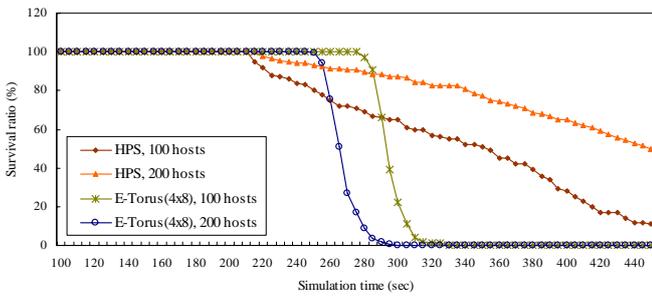


Fig. 10. Survival ratios of the proposed protocol (HPS) and quorum-based asynchronous PS (QAPS) protocol (using the Torus(4x8) quorum system [7]). (100 ~ 200 hosts, mobility = 10m/sec)

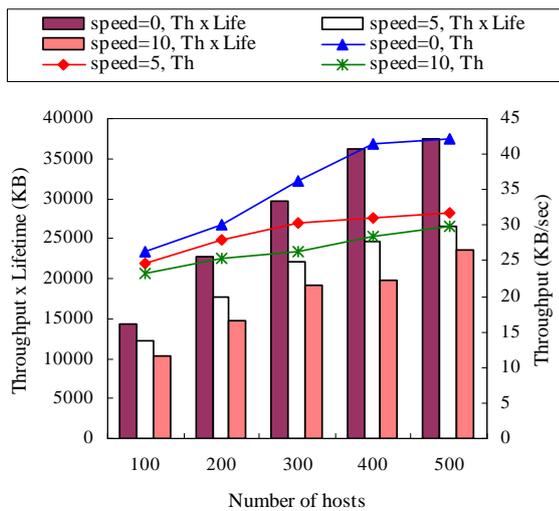


Fig. 11. Throughput and throughput×lifetime of the proposed protocol for different speeds and host densities. (number of hosts = 100 ~ 500, mobility = 0 ~ 10m/sec)

Fig. 11 shows the impact of mobility on throughput and aggregate throughput (throughput×lifetime) for the

proposed protocol. We observe that mobility has a negative impact on both metrics because more retransmissions are incurred as hosts move faster. However, as shown in Fig. 12, the proposed protocol is better than the QAPS protocol (using torus(4×8) quorum system) in both metrics. For the comparison sake, we also show the performance of the non-power saving protocol, which is denoted as AA (always active) protocol. The throughput of the AA protocol is the best since all the hosts running the protocol always turn their radio on. However, the AA protocol is far worse than the proposed protocol in terms of aggregate throughput. This is because the proposed protocol has much longer system lifetime than the AA protocol.

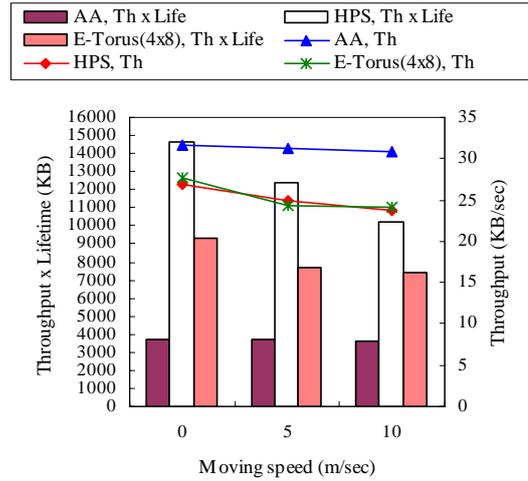


Fig. 12. The comparison of the proposed protocol (HPS), the quorum-based asynchronous PS (QAPS) protocol (using Torus(4x8) quorum system) and the non-power-saving (AA, always active) protocol in terms of throughput and throughput×lifetime. (100 hosts, mobility = 0 ~ 10m/sec)

V. CONCLUSION

In this paper, we have proposed a hybrid PS protocol for an IEEE 802.11-based MANET to take advantages of both the synchronous and the asynchronous PS protocols to save more energy and to accommodate more hosts. The protocol utilizes the concept of dual-channel and dual-transmission-range clustering. It divides all the hosts into clusters. Each cluster has a cluster head and all the heads are organized as a virtual backbone to help route data. The synchronous PS protocol is operated in an individual cluster, and the quorum-based asynchronous PS protocol is operated among cluster heads. The proposed protocol also uses the cluster head dismissal mechanism to void the ever increasing of cluster heads, and is thus adaptive to topology changing. We have shown by simulation that the proposed protocol is more power-efficient and more scalable than related protocols.

The IEEE 802.11b/g standards and IEEE 802.11a standard provide 3 and 12 non-overlapped frequency channels, respectively [13]. All the standards also provide the capability of adjusting radio transmitting power. These features of the standards make the proposed protocol practical. It is noted that the proposed protocol does not demand hosts to equip two transceivers although it utilizes the dual-channel concept. Therefore, the proposed protocol is applicable to MANETs composed of hosts with single

IEEE 802.11 network interface card.

ACKNOWLEDGEMENT

This work is sponsored by NSC of Taiwan under grant-number NSC93-2213-E008-046.

REFERENCES

- [1] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks," *Proc. of the International Conference on Mobile Computing and Networking*, pages 85–96, 2001.
- [2] C. F. Chiasserini and R. R. Rao, "A Distributed Power Management Policy for Wireless Ad Hoc Networks. IEEE Wireless Communication and Networking Conference, pages 1209-1213, 2000.
- [3] J. Gomez, A. T. Campbell, M. Naghshineh, and C. Bisdikian, "A Distributed Contention Control Mechanism for Power Saving in random-access Ad-Hoc Wireless Local Area Networks," *Proc. of IEEE International Workshop on Mobile Multimedia Communications*, pages 114-123, 1999.
- [4] J. C. Haartsen, "The Bluetooth Radio System," *IEEE Personal Communications*, pages 28-36, Feb 2000.
- [5] L. Hu, "Topology Control for Multihop Packet Radio Networks," *IEEE Transactions on Communications*, 41:1474-1481, Oct 1993.
- [6] C. F. Huang, Y. C. Tseng, S. L. Wu, and J. P. Sheu, "Increasing the Throughput of Multihop Packet Radio Networks with Power Adjustment," *International Conference on Compute, Communication, and Networks*, 2001.
- [7] J.-R. Jiang, Y.-C. Tseng, C.-S. Hsu, and T.-H. Lai, "Quorum-Based Asynchronous Power-Saving Protocols for IEEE 802.11 Ad Hoc Networks," *ACM Journal on Mobile Networks and Applications*, Feb. 2005
- [8] E.-S. Jung and N. H. Vaidya, "An Energy Efficient MAC Protocol for Wireless LANs," *IEEE INFOCOM*, 2002.
- [9] LAN MAN Standards Committee of the IEEE Computer Society, IEEE Std 802.11-1999, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications," *IEEE*, 1999.
- [10] J. R. Lorch and A. J. Smith, "Software Strategies for Portable Computer Energy Management," *IEEE Personal Communications*, pages 60-73, Jun 1998.
- [11] C. E. Perkins and E. M. Belding-Royer, "Ad-hoc on-demand distance vector routing," *IEEE Workshop on Mobile Computing Systems and Applications*, pages 90–100, 1999.
- [12] R. Ramanathan and R. Rosales-Hain, "Topology Control of Multihop Wireless Networks using Transmit Power Adjustment," *IEEE INFOCOM*, pages 404-413, 2000.
- [13] Ashish Raniwala and Tzi-cker Chiueh, "Architecture and Algorithms for an IEEE 802.11-based Multi-channel Wireless Mesh Network," *IEEE INFOCOM*, 2005.
- [14] J. H. Ryu and D. H. Cho, "A New Routing Scheme Concerning Power-Saving in Mobile Ad-Hoc Networks," *Proc. of IEEE International Conference on communications*, 3:1719–1722, 2000.
- [15] J. H. Ryu, S. Song, and D. H. Cho, "A Power-Saving Multicast Routing Scheme in 2-tier Hierarchical Mobile Ad-Hoc Networks," *Proc. of IEEE Vehicular Technology Conference*, 4:1974–1978, 2000.
- [16] A. K. Salkintzis and C. Chamzas, "An In-Band Power-Saving Protocol for Mobile Data Networks," *IEEE Transactions on Communications*, 46:1194-1205, Sep 1998.
- [17] T. Simunic, H. Vikalo, P. Glynn, and G. D. Micheli, "Energy Efficient Design of Portable Wireless Systems," *Proc. of the International Symposium on Low Power Electronics and Design*, pages 49-54, 2000.
- [18] S. Singh and C. S. Raghavendra, "Power Efficient MAC Protocol for Multihop Radio Networks," *Proc. of IEEE International*

Personal, Indoor and Mobile Radio Communications Conference, pages 153-157, 1998.

- [19] S. Singh, M. Woo, and C. S. Raghavendra, "Power-Aware Routing in Mobile Ad Hoc Networks," *Proc. of the International Conference on Mobile Computing and Networking*, pages 181-190, 1998.
- [20] Y. C. Tseng, C. S. Hsu, and T. Y. Hsieh, "Power-Saving Protocols for IEEE 802.11-Based Multi-Hop Ad Hoc Networks," *IEEE INFOCOM*, 2002.
- [21] R. Wattenhofer, L. Li, P. Bahl, and Y. M. Wang, "Distributed Topology Control for Power Efficient Operation in Multihop Wireless Ad Hoc Networks," *IEEE INFOCOM*, pages 1388–1397, 2001.
- [22] H. Woesner, J. P. Ebert, M. Schlager, and A. Wolisz, "Power-Saving Mechanisms in Emerging Standards for Wireless LANs: The MAC Level Perspective," *IEEE Personal Communications*, pages 40-48, Jun 1998.
- [23] S. L. Wu, Y. C. Tseng, and J. P. Sheu, "Intelligent Medium Access for Mobile Ad Hoc Networks with BusyTones and Power Control," *IEEE Journal on Selected Areas in Communications*, 18:1647-1657, Sep 2000.
- [24] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed Energy Conservation for Ad Hoc Routing," *Proc. of the International Conference on Mobile Computing and Networking*, pages 70-84, 2001.



Jehn-Ruey Jiang received his Ph. D. degree in Computer Science in 1995 from National Tsing-Hua University, Taiwan, R.O.C. He joined Chung-Yuan Christian University as an Associate Professor in 1995. He joined Hsuan-Chuang University in 1998 and became a full Professor in 2004. He is currently with the Department of Computer Science and Information Engineering, National Central University. He is a recipient of the Best Paper Award in 32nd International Conference on Parallel Processing, 2003, and a recipient of Excellent Paper Award in Mobile Computing 2004. His research interests include distributed computing, mobile computing, peer-to-peer computing, pervasive computing, distributed fault-tolerance, protocols for mobile ad hoc networks and wireless sensor networks.



Chau-Yuan Yang received the BE and ME degree in 1996 and 1998 from the Department of Computer Science and Engineering, Yuan Ze University, Taiwan. He is currently a PhD student at the Department of Computer Science and Information Engineering, National Central University, Taiwan. His research interests include Self-Stabilizing System and Wireless Sensor Networks.



Ting-Yao Chiou received the B.B.A. degree in Information Management from the Fu Jen Catholic University, Hsinchuang, Taiwan, in June 2004. He is currently a graduate student in computer science and information engineering at the National Central University. His research interests include wireless networks, wireless sensors networks



Shing-Tsaan Huang is a fellow of the IEEE. He received the PhD degree in 1985 from the Department of Computer Science, University of Maryland at College Park. Before he pursued the PhD degree, he worked several years in the computer industry in Taiwan. Dr. Huang is currently a professor at the Department of Computer Science and Information Engineering, National Central University, Taiwan. He also serves as the vice president of the university. His research interests include self-stabilizing systems and distributed computing.

and distributed computing.