A Reliable and Efficient MAC Layer Broadcast (Multicast) Protocol for Mobile Ad Hoc Networks

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

*Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan.

†Department of Computer Science and Information Engineering, National Central University, Taipei, Taiwan.

Abstract—**Broadcast/multicast is a key service for mobile ad hoc networks. A great number of applications rely on a reliable and efficient MAC layer broadcast. The IEEE 802.11 broadcast protocol, which is based on Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA), does not offer any MAC layer recovery on broadcast frames. Consequently, the increasing probability of lost frames may deteriorate the quality of broadcast/multicast services offered at upper layers. Previous protocols extended Request-To-Send (RTS), Clear-To-Send (CTS) and Acknowledgement (ACK) to enhance the broadcast reliability. However, they brought about the hidden terminal problem and the excessive retransmission problem at the same time. In this paper, we first formulate the broadcast problem as an optimization problem and show that it is NP-hard even if the upper layer service is periodical beacons. An approximation algorithm with a guaranteed approximation ratio is also suggested. Then a reliable and efficient MAC layer broadcast protocol, named Broadcast Protocol with Busy Tone (BPBT), is proposed. BPBT applies a busy tone to solve the hidden terminal problem. Finally, BPBT is compared with previous protocols for performance evaluation by simulation.**

Keywords-Ad hoc network; approximation algorithm; broadcast; busy tone; MAC

I. INTRODUCTION

In recent years, explosive growth of wireless communications has offered various services and conveniences to human being. Plenty of these modern services can be delivered via the deployment of wireless accesses with fixed networks as their backbones. However, there are several applications such as disaster rescues, tactical communications for military usages and wireless conference meetings, for which it is hard or wasteful to establish a fixed communication infrastructure. Instead, a temporary ad hoc network can serve for them. An ad hoc network can provide communication services for a collection of mobile nodes without any infrastructure or centralized access point. In a multi-hop ad hoc network, each mobile node can act as a router to relay data packets to their neighboring mobile nodes.

Broadcast/multicast is a key service for mobile ad hoc networks. A great number of applications require a reliable and efficient Medium Access Control (MAC) layer broadcast. They include, for example, periodical beacons, alarm signals to all nodes, route discovery phases in on-demand routing protocols, and multicasting video streams. Many broadcast/multicast protocols have been proposed in the literature [1]-[13], and some (see [1]-[6]) of them are devoted to the network layer. The execution of these network layer protocols relies heavily on a reliable and efficient MAC layer broadcast. However, the IEEE 802.11 broadcast protocol [7], which is based on Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA), does not offer any MAC layer recovery on broadcast frames. Consequently, the increasing probability of lost frames may deteriorate the quality of broadcast/multicast services offered at upper layers.

There are also broadcast protocols that are executed at the MAC layer. They include BSMA [8], BMW [9], BMMM [10], LAMM [10], and others (see [11]-[13]). In [8], BSMA extended the IEEE 802.11 control frames: Request-To-Send (RTS), Clear-To-Send (CTS) and Negative AcKnowledgement (NAK), to enhance the reliability of broadcast/multicast services. However, BSMA was not able to coordinate the transmission of CTS frames, which brought about a serious problem: the collision of CTS frames. Although the broadcast protocols proposed in [9]-[13] could avoid the collision problem, they induced another problem, the hidden terminal problem [14], at the same time. They are briefly described below.

Each CTS frame in unicast not only notifies the sender that the receiver is ready to receive data frames, but also prevents hidden terminals from transmission. On the other hand, in order to avoid the collision of CTS frames in broadcast, CTS frames are either transmitted at different times [10], or replaced with a CTS frame from a leader node [9], [12]. Both methods fail to prevent hidden terminal from transmission. The broadcast protocols proposed in [11], [13] used different methods to avoid the collision problem. They also fail to prevent hidden terminals from transmission.

The broadcast protocols in [8]-[13] may suffer from another serious problem: excessive retransmission of data frames, as explained as follows. These protocols all used ACKnowledgement (ACK) or NAK frames to enhance the reliability of broadcast/multicast services. If the sender does not receive an ACK (or receives an NAK) after sending out a data frame, it will retransmit the data frame. Since these protocols also suffer from the hidden terminal problem, collision of data frames may happen frequently, which prevents the receiver from sending an ACK or trigger the receiver to send an NAK.

Although heavy collision of data frames will deteriorate the quality of broadcast/multicast services, a broadcast protocol without any collision is not the best solution. An interesting fact that moderate collision is helpful to the quality of broadcast/multicast services is found in this paper. Different broadcast protocols (with moderate collision) are required for various broadcast/multicast services to optimize their performances. In this paper, we first design an optimal broadcast protocol for periodical beacon. Then we adapt the protocol for other broadcast/multicast services.

Broadcast for periodical beacon can be formulated as an optimization problem. We show that the optimization problem is NP-hard [15] and then solve it by an approximation algorithm. The approximation ratio of the approximation algorithm is estimated. Based on the obtained approximation solution, an efficient and reliable MAC layer broadcast protocol, named Broadcast Protocol with Busy Tone (BPBT), is then proposed. BPBT can avoid the collision of CTS frames. Besides, BPBT applies the busy tone to avoid the hidden terminal problem. The busy tone was used before to avoid the hidden terminal problem in unicast (see [14], [16], [17]). BPBT has the same approximation ratio as the approximation algorithm if the broadcast/multicast service offered at the network layer is periodical beacon. The performance of BPBT is also evaluated by simulation.

The remainder of this paper is organized as follows. In Section II, the broadcast problem defined at the MAC layer is introduced. Broadcast for periodical beacon is formulated as an optimization problem, and an approximation algorithm for solving it is presented. Broadcasts for other services are also discussed. In Section III, BPBT is elaborated as a consequence of Section II. In Section IV, simulation results, which compare BPBT with previous broadcast protocols, are shown. Finally, in Section V, this paper is concluded with some remarks.

II. BROADCAST/MULTICAST PROBLEM AT MAC AND NETWORK LAYER

A. Broadcast (Multicast) Problem at MAC Layer

The definition of multicast problem at the MAC layer is as follows. A source node intends to send data frames successfully to some of its neighboring nodes. Broadcast is a special case of multicast. According to our definition, if a sender s wants to transmit data frames, the state of its intended receivers must be *Ready*, *Busy* or *Receiving*. The state of a receiver v is *Receiving* meaning that v is receiving data frames from other sender s'. If s transmits data frames to v, s and s' will retransmit their data frames resulting from collisions. The state of a receiver v is *Busy* meaning that v does not receive data frames from other senders, but it also cannot receive data frames from s. This situation results from v transmitting or located at other senders' coverage. If s transmits data frames to v, s will retransmit its data frames resulting from collisions. Finally, the state of a receiver v is *Ready* meaning that v can receive data frames from s without collision.

Recall that the RTS/CTS handshake used in [8]-[10] may cause excessive retransmission of data frames. It is further explained as follows. Once receiving an RTS from the sender, a *Ready* intended receiver will reply a CTS, whereas a *Receiving* or *Busy* intended receiver will keep silent. The sender will start transmission if it gets a CTS, and collision will happen if one or more of its intended receivers are *Receiving* or *Busy*. Retransmission will succeed only when all intended receivers are *Ready*.

As described above, when the sender transmits data frames to *Receiving* receivers, collision will happen. Such collision costs highly because it brings about more retransmissions. In order to avoid such inefficiency, the sender had better transmit data frames only when the intended receivers are *Read*y or *Busy*.

B. Broadcast/Multicast Problem at Network Layer

There are a number of broadcast/multicast services at the network layer which require efficient and reliable broadcasts at the MAC layer. They include, for example, periodical beacons, alarm signals to all nodes, route discovery phases in on-demand routing protocols, and multicasting video streams. In order to optimize their performances, different broadcast protocols (at the MAC layer) will be designed. In [18], considering an alarm signal to all nodes the upper layer service, the problem of designing a broadcast protocol to minimize the bandwidth consumption or time delay was proved NP-hard.

In this section, another service, i.e., periodical beacons, is considered. In many routing/multicasting protocols (e.g., see [19], [20]), beacons were emitted periodically to update routing/multicasting tables. Suppose that there are a set of nodes in the ad hoc network which are required to send beacons to their neighbors. We intend to design a broadcast protocol to minimize the completion time. The ad hoc network can be conveniently represented by an undirected graph $G=(V, E)$, where each vertex in V uniquely corresponds to a node and each edge (u, v) in E denotes that u and v are two neighboring nodes. Assume that the time axis is divided into continuous time slots, denoted by s_i, where $i \geq 0$. The problem can be formally expressed as follows.

Let $V' \subset V$ be the set of nodes that are required to send beacons to their neighbors, $S=\{s_0, s_1, s_2, \ldots\}$, and T be a mapping from V' to 2^S (the power set of S). The meaning of T is as follows: $T(u) \in 2^S$ contains s_i if and only if u broadcasts a beacon at the beginning of s_i. Also let $N(u)$ denote the set of neighboring nodes of u. T is *feasible* if for every $u \in V'$, $v \in N(u)$ and $v' \in N(v) \cup \{v\}$, $T(u)$ is not a subset of $T(v')$. The problem is to determine an optimal feasible mapping, denoted by T^*, so that $\max\{i \mid s_i \in T^*(u)$ for some $u \in V'\}$ is minimized.

THEOREM 1. The problem of determining T^* is NP-hard.

Proof. To prove this theorem, we reduce an NP-complete problem, i.e., the 3-coloring problem to our problem. Given an undirected graph $G=(V, E)$, the 3-coloring problem [21] inquires if G can be colored with three distinct colors so that no two adjacent vertices have the same color. In other words, we want to know if there exists a mapping F from V to $\{0, 1, 2\}$ so that $F(u) \neq F(v)$ for every $(u, v) \in E$.

Suppose that $G=(V, E)$ is an arbitrary instance of the 3-coloring problem. The following shows a reduction from G to an instance, denoted by $G_b=(V_b, E_b)$ and $V_b^{'}$, of our problem, where $V_b^{'} \subset V_b$ is the collection of nodes required to broadcast beacons.

- $V_b=V+M$, $\forall u, v \in V$, if there is an edge $(u, v) \in E$, then there is a node $m_{u,v} \in M$.
- $\forall u, v \in V$, if there is an edge $(u, v) \in E$, then there are two edges, $(u, m_{u,v})$, $(v, m_{u,v}) \in E_b$.
- $V_b^{'}=V$.


```
Strict_Mode_Procedure ( Input: G(V, E), V', Output: T(v), ∀v∈ V' )
    time_slot = 1
    U = V'
    while U ≠ Φ
        for each v∈ U
            if ∀u∈ N_G*(v), T(u) ≠ time_slot
                T(v) = time_slot
                U = U – {v}
        time_slot = time_slot + 1
```

Fig. 1. The proposed approximation algorithm.

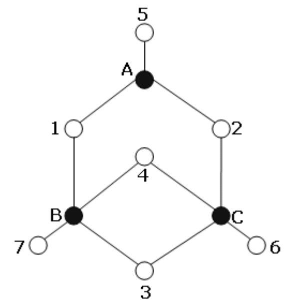

Fig. 2. An example for greedy, loose, strict schemes.

If T^* of the instance of our problem can be determined, then $\max\{i \mid s_i \in T^*(u)$ for some $u \in V'\}$, denoted by s^*, can be obtained. If s^* is less than 3, $T^*(u)$ must be the one of following sets, $\{s_0\}$, $\{s_1\}$, $\{s_2\}$, $\{s_0, s_1\}$, $\{s_0, s_2\}$, $\{s_1, s_2\}$ and $\{s_0, s_1, s_2\}$, for all $u \in V_b^{'}$. Since T^* is a feasible mapping, $\forall m_{u,v} \in M$, the one of $T^*(u)$ and $T^*(v)$ is not a subset of another one. If $T^*(u)$ is a subset of a $T^*(v)$ (or the opposite case), collisions will occur at $m_{u,v}$ and u (or v) does not broadcast to $m_{u,v}$ successfully in three time slots. According to this specific, if the $T^*(u)$ of a node u is $\{s_0, s_1, s_2\}$, the node u must be an isolated node in G and can be ignored in the 3-coloring problem. When a node v is a neighboring node of a node u in G and $T^*(u)$ is $\{s_0, s_1\}$, $T^*(v)$ must be the one of $\{s_2\}$, $\{s_0, s_2\}$ and $\{s_1, s_2\}$. The rest may be deduced by analogy. Then we can find another mapping T' by the following way.

- If $T^*(u)$ is $\{s_0, s_1\}$, $T'(u)$ is $\{s_0\}$.
- If $T^*(u)$ is $\{s_1, s_2\}$, $T'(u)$ is $\{s_1\}$.
- If $T^*(u)$ is $\{s_0, s_2\}$, $T'(u)$ is $\{s_2\}$.
- If $T^*(u)$ is $\{s_0\}$, $\{s_1\}$ or $\{s_2\}$, $T'(u)=T^*(u)$.

Obviously, T' is also a feasible mapping and $T'(u)$ must be the one of $\{s_0\}$, $\{s_1\}$ and $\{s_2\}$ for all $u \in V_b^{'}$. Therefore, we can find a mapping F that $F(u)=i$ if $T'(u)=\{s_i\}$ for all $u \in V_b^{'}$ and $i \in \{0, 1, 2\}$ so that $F(u) \neq F(v)$ for every $(u, v) \in E$. ①

If s^* is more than or equal to 3, we can prove simply that the original graph G can not be colored by three colors. By contradiction, if G can be three colored, the corresponding solution for coloring method is a feasible solution of our problem and s^* is less than 3. ②

According to ① and ②, we can know that our problem is harder than the 3-coloring problem. Since the 3-coloring problem is NP-complete, our problem is NP-hard. **Q.E.D.**

Since the construction of optimal algorithms is not practical, we resort to approximate solutions. In above discussion, obviously, if the number of nodes that want to transmit a broadcast data frame to a node v is $d'(v)$, the minimum number of time slots required for this problem is $d'(v)$ at least. If $v \in V'$, the minimum number of time slots required for this problem is $d'(v)+1$ at least. We denote this value of v as $D'(v)$ and the maximum $D'(v)$ for all $v \in V$ as $MS(G, V')$. $MS(G, V')$ is the lower bound of this problem. In the following description, an approximation algorithm would be designed, and the upper bound of this algorithm would be derived and proved in Theorem 2. Then, the approximation ratio of this algorithm can be obtained from the lower and upper bound.

The main idea of this algorithm is greedily selecting nodes to broadcast without any collision in each time slot. The detailed procedure is showed in Fig. 1. In Fig. 1, G^* is a graph that is $G^2 - (V - V')$. G^* represents a collision relation between all nodes of V'. If two nodes that are the neighbors of each other in G^* broadcast at same time slot, collisions will occur. $N_{G^*}(v)$ is a set of nodes that are neighbors of node v in G^*.

THEOREM 2. The upper bound of the result of the above algorithm is $MD(G^*)+1$. $MD(G)$ is the maximum node degree of G.

Proof. Since no collision occur, each node of V' broadcasts once. Obviously, the set of nodes broadcasting at slot i is the maximal independent set of $G^* - (V' - U_i)$ in this algorithm. U_i is the set of nodes that do not broadcast after i-1 time slots. By contradiction, we assume that there is a node v broadcasting after $MD(G^*)+1$ time slots. Since the degree of v in G^* is less than $MD(G^*)+1$, there is a slot $t \leq MD(G^*)+1$ and all neighboring nodes of v in G^* do not broadcast at slot t. The set of nodes broadcasting at slot t is not the maximal independent set because the node v can be selected to broadcast at t.

Q.E.D.

C. Broadcast for General Services

The approximation ratio that is $\frac{MD(G^*)+1}{MS(G,V')}$ can be obtained by previous discussion. However, periodical beacon is the only one of many broadcast/multicast services. In fact, the complexity of similar problems for other broadcast/multicast services may be NP-complete or NP-hard mostly. So designing a broadcast protocol to optimize performance for various broadcast/multicast services is impracticable. Instead, we resort to more general schemes to satisfy more broadcast/multicast services at network layer. A moment of an active ad hoc network can be regard as a problem that some nodes want to unicast or broadcast frames to some of their neighbors without respecting to what kinds of broadcast/multicast services offered at network layer. This problem is more general than the problem defined in Section II-B and called as general broadcast problem. An interesting fact that moderate collision is helpful to the quality of broadcast/multicast services is found in the following description.

Previous protocols always broadcast data frames when an intended receiver is *Ready*. This broadcast scheme is called as loose scheme. Another extreme scheme is called as strict scheme. It broadcasts data frames when all intended receivers are *Ready*. Obviously, the algorithm referred in Section II-B is the strict scheme. These two schemes do not work well in the

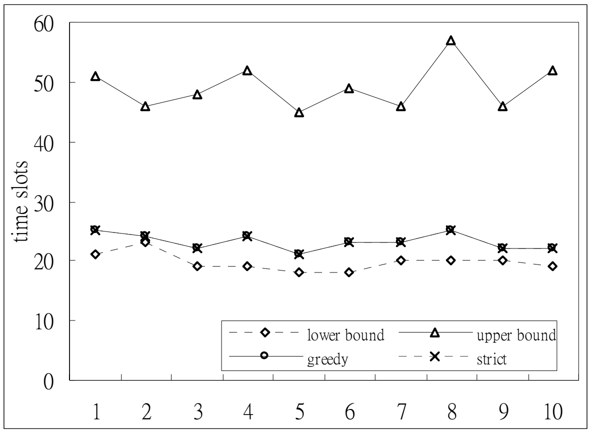

Fig. 3. The upper bound, lower bound, and the number of time slots required for strict, greedy schemes in 10 different graphs.

general broadcast problem since they suffer from serious collisions and many forbidden broadcasts respectively. For example, in Fig. 2, node A intends to transmit a data frame to node 1, 2, 5. Node B intends to transmit a data frame to node 3, 4, 7. Node C intends to transmit a data frame to node 3, 4, 6. Obviously, there must be three time slots at least to finish these broadcasts in the loose scheme. The strict scheme uses exactly three time slots to finish these broadcasts. Each of them is not the optimal solution. The minimum number of time slots required for this case is 2. Node A and B transmit at time slot 1, and then A and C transmit at time slot 2.

Recall that transmitting data frames from a sender to *Receiving* intended receivers will incur collision of higher cost than transmitting data frames from a sender to *Busy* intended receivers. It is natural to avoid transmitting data frames to *Receiving* intended receivers. According to this idea, a broadcast scheme, called as greedy scheme, has been proposed. In greedy scheme, a node will broadcast its data frames when all of its neighboring nodes are not *Receiving* and there is an intended receiver that is *Ready* at least.

We randomly construct a connected graph with 110 nodes. Fig. 3 shows the upper bound, lower bound, and the number of time slots required for each above schemes in 10 different graphs when the broadcast/multicast service is periodical beacon. The behavior of the greedy scheme is similar to the strict scheme if intended receivers of each node are all their neighbors. So the greedy scheme has the same approximation ratio as the strict scheme in the periodical beacon problem. Moreover, Fig. 4 shows that the performance of the greedy scheme is better than the original approximation algorithm in other broadcast/ multicast services. The multicast ratio is the ratio of the number of intended receivers to the number of all neighbors. If the multicast ratio is 1, it means that all neighbors are intended receivers of each node. In the following section, the detailed procedure of implementing the BPBT according to the greedy scheme will be described.

III. Broadcast Protocol with Busy Tone

In previous discussion, each node must know the detailed

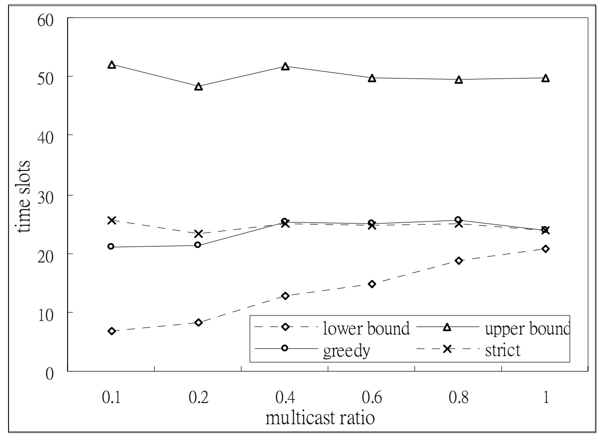

Fig. 4. The average value of upper bound, lower bound, and the number of time slots required for strict, greedy schemes with different multicast ratios.

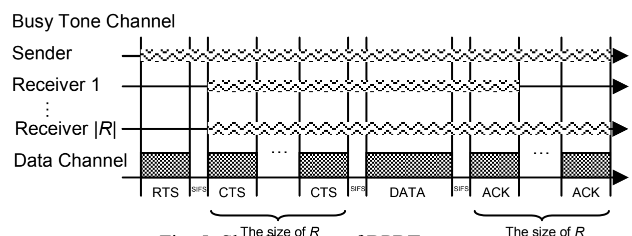

Fig. 5. Slot diagrams of BPBT.

information of states of all its neighboring nodes to perform greedy scheme. However, a sender can not distinguish that an intended receiver is *Busy* or *Receiving* when this receiver does not return CTS. So BPBT uses a narrow-band busy tone to notify neighboring nodes of the on-going communication. By the aid of busy tone, senders can distinguish that each intended receiver is *Busy* or *Receiving*.

We assume that each node has two antennas. One antenna is required for transmitting and receiving the busy tones while another antenna is for transmitting and receiving data frames. In BPBT, a node wishing to transmit senses the busy tone channel instead of common channel and if it is free for a time equal to the DCF InterFrame Space (DIFS), the node performs the RTS/CTS handshake. If any CTS return, the node transmits. Fig. 5 shows the slot diagrams of BPBT and the detailed procedure is as follows.

When a source node *s* has a data frame to send to a set of intended receivers *R*:

1) If busy tone channel is free for a time equal to the DIFS, *s* turns on its busy tone and transmits a control frame (RTS) which indicates a back-off time of each intended receiver of *R*. Otherwise, *s* suspends its transmission and continues to sense the busy tone channel.

2) An intended receiver which has received the RTS turns on its busy tone and replies CTS to *s* according its back-off time respectively.

3) The sender *s* waits a time period for receivers' CTS and then broadcasts the data frame for any replied CTS. If there is no CTS replied after time expiring, *s* backs off and goes to step 1.

4) An intended receiver which has received the data frame transmits ACK to *s* according its back-off time respectively and then turns off its busy tone.

5) If *s* has received ACKs from all intended receivers of *R*, the broadcast is complete. Otherwise, assuming *R*' is a set of intended receivers from which *s* has received an ACK and then setting *R* is *R*-*R*'. Go to step 1.

In the procedure of BPBT, a node will turn on its busy tone when it is receiving. If a node can pass through the collision avoidance phase of BPBT, each neighbor of it must be *Busy* or *Ready* only. Then BPBT applies RTS/CTS handshake to distinguish that the intended receivers is *Busy* or *Ready*. Any intended receiver being *Ready* implies that the number of successful receivers will increase if the sender broadcasts.

On the other hand, the problem referred in [10] must be discussed again in this protocol. According to IEEE 802.11, after the medium is idle for a time equal to DIFS, every node can contend for access to the wireless channel. So the number of intended receivers should be less than 2. But this problem does not exist in BPBT. This is because that the collision avoidance phase in BPBT is listening the busy tone channel. When an intended receiver defers its CTS or ACK, its busy tone does not turn off. So no nodes can contend for access to the wireless channel to break the CTS or ACK of the on-going communication.

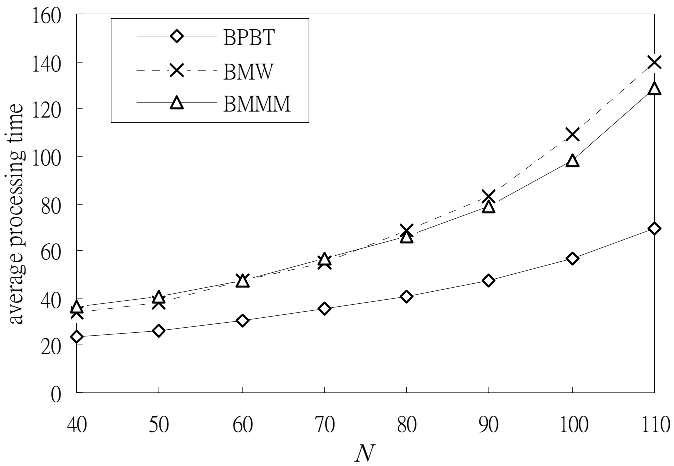

Fig. 6. Average processing times of periodical beacons with different values of *N*.

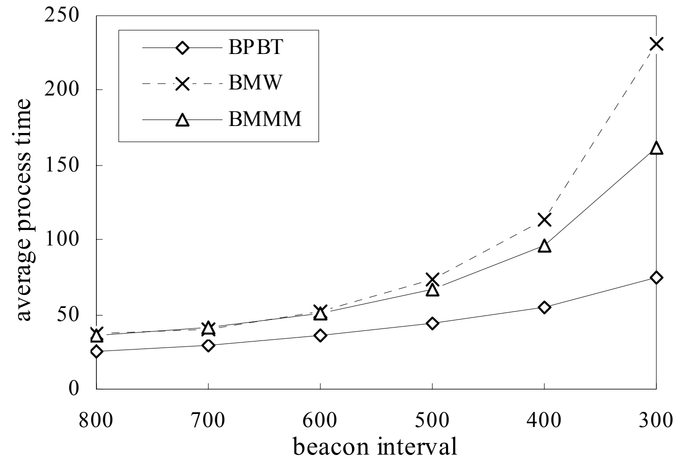

Fig. 7. Average processing times of periodical beacons with different beacon intervals.

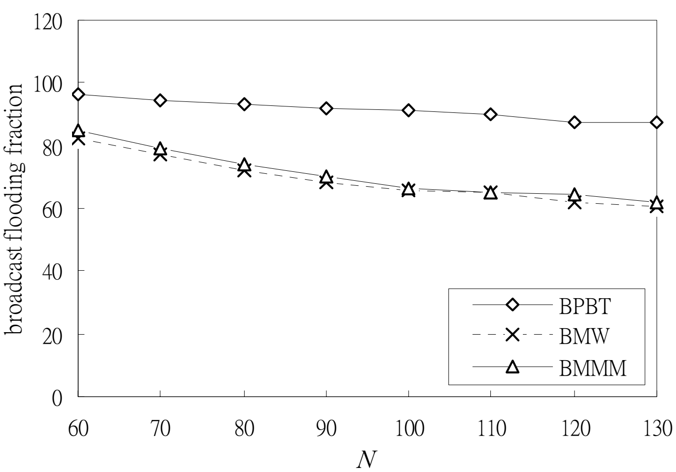

Fig. 8. Broadcast flooding fractions of alarm signals to all nodes with different values of *N*.

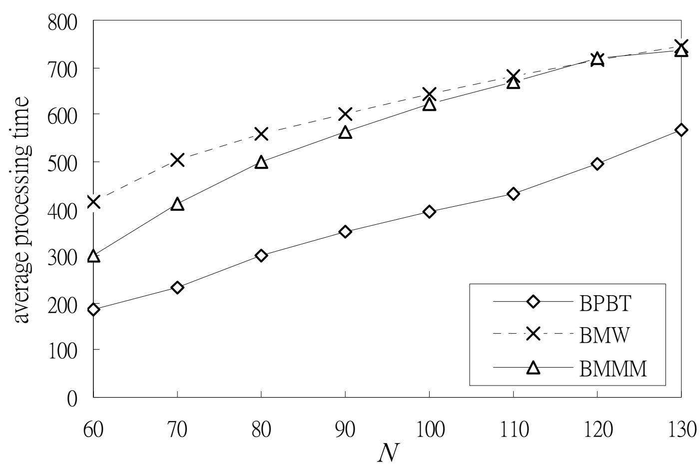

Fig. 9. Average processing times of alarm signals to all nodes with different values of *N*.

IV. Simulation Results

A mobile ad hoc network of *N* nodes which were randomly spread in a 400m×400m area was simulated, where *N* is a parameter. Nodes were free to move anywhere within this area. The radio transmission range of each sending node was assumed 80 meters. The time axis was divided into continuous time slots and events happened at the beginning of time slots. The transmission time for a data frame (a control frame) is 10 time slots (1 time slot). The simulation continued for 5000000 time slots. Each node randomly decided to take a migration or rest. If it decided to migrate, it would travel towards a random spot with a constant speed that was randomly determined from 2 to 8 meters per 5000 time slots. If it decided to rest, it would randomly choose a rest period from 300000 to 1500000 time slots. When it finally reached the spot or exhausted the rest period, it would decide to migrate or rest again.

A. Periodical Beacons

All nodes broadcast beacons periodically to all of their neighbors throughout the simulation. The processing time for a node to broadcast a beacon is the consuming time to transmit the beacon successfully to all of its neighbors. Fig. 6 and Fig. 7 showed the average processing times required for BPBT, BMW and BMMM with different values of N and different beacon intervals, respectively, where each average processing time took the average of all processing times required for all nodes to complete their beacon transmissions. The beacon interval in Fig. 6 was assumed 1000 time slots and N=50 was assumed in Fig. 7. As observed from Fig. 6 and Fig. 7, BPBT has a better performance than BMW and BMMM, which is a consequence of BMW and BMMM suffering from the hidden terminal problem and the excessive retransmission problem.

B. Alarm signals to all nodes

Some nodes were asked to broadcast packets to all nodes by flooding. The broadcast frequency was assumed once per 1000 time slots. That is, one node was randomly selected to broadcast every 1000 time slots. The number of retransmissions (retry count) for each sending node was at most three, even if not all intended receivers had received the broadcast frames. Performance metrics we adopted were the broadcast flooding fraction and the average processing time of all broadcast requests. The *broadcast flooding fraction* is the ratio of the number of nodes having the broadcast packets to the total number of nodes.

Fig. 8 and Fig. 9 compared the broadcast flooding fraction and average processing time, respectively, with respect to BPBT, BMW and BMMM when the value of N varied. BPBT was superior to BMW and BMMM in both metrics because BMW and BMMM had more frequent collisions. In fact, average processing times of BMW and BMMM will grow exponentially if the retry count is unlimited.

V. CONCLUSION

A broadcast protocol, Broadcast Protocol with Busy Tone (BPBT) used busy tone to solve the hidden terminals problem. Moreover, BPBT has an approximation ratio if the broadcast/multicast service offered at the network layer is periodical beacon. Simulation was made for comparing BPBT with BMW and BMMM, and simulation results showed that BPBT is superior to BMW and BMMM in average processing time and broadcast flooding fraction. This is a consequence of BPBT having a better scheduling policy and not suffering the hidden terminal problem.

Acknowledgement

This work was supported by the MediaTek Inc. under the project "Wireless Communication Systems". This work was also supported by National Science Council of Taiwan under the NSC93-2524-S-008-002 Project.

REFERENCES

[1] Y. C. Tseng, S. Y. Ni, Y. S. Chen, and J. P. Sheu, "The broadcast storm problem in a mobile ad hoc network," *ACM Wireless Network*, vol. 8, no. 2/3, pp. 153-167, March, 2002.

[2] G. Calinescu, I. Mandoiu, P. J. Wan, and A. Zelikovsky, "Selecting forwarding neighbors in wireless ad hoc networks," *ACM Mobile Networks and Applications*, vol. 9, no. 2, pp. 101-111, April, 2004.

[3] W. Liang, "Constructing minimum-energy broadcast trees in wireless ad hoc networks," *Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking and Computing* (*MOBIHOC*), pp. 112-122, June, 2002.

[4] H. Lim and C. Kim, "Multicast tree construction and flooding in wireless ad hoc networks," *Proceedings of the 3rd ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems* (*MSWIM*), pp. 61-68, August, 2000.

[5] S. J. Lee, M. Gerla, and C. C. Chiang, "On-demand multicast routing protocol," *Proceedings of the IEEE Wireless Communications and Networking Conference* (*WCNC*), pp. 1298-1302, September, 1999.

[6] J. J. Garcia-Luna-Aceves and E. L. Maddruga, "The core-assisted mesh protocol," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 1380-1394, August, 1999.

[7] IEEE 802.11 Working group, "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," *IEEE Standard 802.11*, September, 1999.

[8] K. Tang and M. Gerla, "Random access MAC for efficient broadcast support in ad hoc networks," *Proceedings of the IEEE Wireless Communications and Networking Conference* (*WCNC*), pp. 454-459, September, 2000.

[9] K. Tang and M. Gerla, "MAC reliable broadcast in ad hoc networks," *Proceedings of the IEEE Military Communications Conference* (*MILCOM*), pp. 1008-1013, October, 2001.

[10] M. T. Sun, L. Huang, A. Arora, and T. H. Lai, "Reliable MAC layer multicast in IEEE 802.11 wireless networks," *Proceedings of the IEEE International Conference on Parallel Processing* (*ICPP*), pp. 527-536, August, 2002.

[11] S. T. Sheu, Y. Tsai, and J. Chen, "A highly reliable broadcast scheme for IEEE 802.11 multi-hop ad hoc networks," *Proceedings of the IEEE International Conference on Communications* (*ICC*), pp. 610-615, May, 2002.

[12] J. Kuri and S. K. Kasera, "Reliable multicast in multi-access wireless LANs," *Proceedings of the IEEE International Conference on Computer Communication* (*INFOCOM*), pp. 760-767, March, 1999.

[13] S. K. S. Gupta, V. Shankar, and S. Lalwani, "Reliable multicast MAC protocol for wireless LANs," *Proceedings of the IEEE International Conference on Communications* (*ICC*), pp. 93-97, May, 2003.

[14] F. Tobagi and L. Kleinrock, "Packet switching in radio channels: part II – the hidden terminal problem in carrier sense multiple-access modes and the busy-tone solution," *IEEE Transactions on Communications*, vol. 23, no. 12, pp. 1417-1433, December, 1975.

[15] M.R. Garey and D. S. Johnson, *Computers and Intractability*, Freeman, 1979.

[16] Z. J. Haas and D. Jing, "Dual busy tone multiple access (DBTMA) – a multiple access control scheme for ad hoc networks," *IEEE Transactions on Communications*, vol. 50, no. 6, pp. 975-985, June, 2002.

[17] 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, September, 2000.

[18] I. Chlamtac and S. Kutten, "On broadcasting in radio networks – problem analysis and protocol design," *IEEE Transactions on Communications*, vol. 33, no. 12, pp. 1240- 1246, December, 1985.

[19] C. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance vector routing (DSDV) for mobile computers," *Proceedings of the ACM Special Interest Group on Communications* (*SIGCOMM*), pp. 233-244, October, 1994.

[20] C. W. Wu and Y. C. Tay, "AMRIS: a multicast protocol for ad hoc wireless networks," *Proceedings of the IEEE Military Communications Conference* (*MILCOM*), pp. 25-29, October, 1999.

[21] C. H. Padadimitriou, *Computational Complexity*, Wesley, 1995.