

行政院國家科學委員會專題研究計畫 期中進度報告

隨意網路中有品質保證之傳輸研究(2/3) 期中進度報告(精簡版)

計畫類別：個別型
計畫編號：NSC 95-2221-E-002-070-
執行期間：95年08月01日至96年07月31日
執行單位：國立臺灣大學資訊工程學系暨研究所

計畫主持人：陳健輝

處理方式：期中報告不提供公開查詢

中華民國 96年06月11日

行政院國家科學委員會專題研究計畫成果報告

隨意網路中有品質保證之傳輸研究

QoS Routing in Mobile Ad-Hoc Networks

計畫編號：NSC 94-2213-E-002-124-

執行期限：94年8月1日至97年7月31日

主持人：陳健輝教授 國立台灣大學資訊工程學系

計畫參與人員：林清池，鄭宇淳，曾義憲，翁憲政

中文摘要

為了在下一代的無線網路上對延遲敏感性應用(delay-sensitive applications)，如聲音和影像，能提供有品質保證的傳輸，隨意網路(MANET)必須能夠提供不同等級傳輸服務品質(quality-of-service)之功能，使得所需要的頻寬和延遲能夠被保證。然而到目前為止，要在隨意網路上提供不同等級傳輸服務品質之功能仍然是一個挑戰。這是因為除了要滿足服務品質的需求，還必須要考量到動態的拓樸(dynamic topology)和共享的無線媒介(shared wireless medium)。這兩點特性產生了以下四個需要解決的問題。第一，一個在共享無線媒介的環境下能夠精確計算可用頻寬(available bandwidth)和在非同步的環境下能夠精確計算點對點延遲(end-to-end delay)的評估方法。第二，一個能適應動態環境的分散式繞徑演算法(distributed routing algorithm)。第三，一個能保證所需求資源的資源保留機制(resource reservation mechanism)。第四，一個快速且有效的路由回復(route recovery)方法。

設計具有服務品質的繞路協定的另一個難題在於拓樸網路會經常變動，那些已經存在的狀態資料會因拓樸毀壞而喪失。因此，對一個繞路協定來說，收集最新的狀態資訊在一個經常變動的環境是非常重要的議題。無論採用任何方法去收集最新狀態資訊，都必須仰賴局部廣播或是全域廣播的功能。在本計劃執行的第二年裏，我們研究一套快速且可靠適用於媒介存取控制層之廣播協定(broadcast protocol)。它可以支援鄰近區域資訊的交換和尋找路由的氾濫式廣播(flooding)。鄰近區域資訊，如區域拓樸、流量和移動性，是計算頻寬不可或缺的資訊。

關鍵詞：隨意網路，服務品質，資源保留，繞徑。

In order to provide quality delivery to delay-sensitive applications such as voice and video in the next generation wireless networks, it is imperative that mobile ad-hoc networks (MANETs) have to support quality-of-service (QoS) so that bandwidth and delay requirements can be guaranteed. Thus far, providing QoS in MANETs is still a challenge, because dynamic topology and shared wireless medium should be taken into account, in addition to QoS constraints. They will bring about the following problems to be resolved: First, an estimation method that can calculate the available bandwidth in a shared wireless channel environment and the end-to-end delay in an unsynchronized environment, accurately; Second, a distributed routing algorithm that can adapt to dynamic environments; Third, a resource reservation mechanism that can guarantee required resources; Fourth, an efficient and effective route recovery method.

Dynamic topology of a MANET is another challenge for designing QoS routing protocols. The state information continuously changes due to load fluctuation. So it is imperative for a QoS routing protocol to gather the up-to-date state information in a dynamic environment. No matter what methods are adopted to gather the up-to-date state information, they rely heavily on local or global broadcasting. In the second year of this project, we propose an efficient and reliable MAC layer broadcast protocol that can support the exchange of neighborhood information and the flooding of route discovery as well. Neighborhood information concerning local topology, traffic and mobility is crucial to bandwidth estimation.

Keywords: MANETs, QoS, resource reservation, routing

一、前言

The people's future living environments are emerging, based upon information resource provided by the connections of various communication networks for users. New small devices like Personal Digital Assistants (PDAs), mobile phones, handhels, and wearable computers enhance information processing and accessing capabilities with mobility. Moreover, traditional home appliances, e.g., digital cameras, cooking ovens, washing machines, refrigerators, vacuum cleaners, and thermostats, with computing and communicating powers attached, extend the field to a fully pervasive computing environment [1]. With this in view, modern technologies should be formed within the new paradigm of pervasive computing, including new architectures, standards, devices, services, tools and protocols.

A pervasive computing environment can be expected based on prestigious infrastructured wireless networks and novel infrastructureless mobile ad-hoc networks (MANETs). A MANET is an autonomous system of mobile hosts (nodes) connected by wireless links. The union of the mobile hosts forms a communication network modeled in the form of an arbitrary communication topology. All communication is done over wireless media, typically by radio through the air, without the help of wired base stations. This is in contrast to the well-known single-hop cellular

network. In cellular networks, communication between two mobile hosts completely relies on the wired backbone and fixed base stations. In contrast, a MANET has no such infrastructures and its network topology may dynamically change in an unpredictable manner.

Dynamic topology of a MANET is another challenge for designing QoS routing protocols. The state information continuously changes due to load fluctuation. So it is imperative for a QoS routing protocol to gather the up-to-date state information in a dynamic environment. Existing QoS routing protocols in MANETs can be classified into three categories: table-driven (proactive) [2], [3], on-demand (reactive) [4-9], [5], [6], [7], [8], [9], and hierarchical and hybrid protocols [10]. Table-driven routing protocols attempt to maintain the up-to-date state and routing information among all nodes, which induces lower latency and higher overhead. However, It was shown in [11], [12] that table-driven protocols are more liable to suffer from performance degradation than on-demand protocols because of the stale information.

In contrast, on-demand routing protocols create routes only as desired by source nodes. When a source node requires a route to a destination node, it initiates a flooding-based route discovery process to gather the up-to-date state information and then determines the route. In general, on-demand routing protocols have higher latency and lower overhead than table-driven routing protocols. However, a flooding process is rather time-consuming and bandwidth-consuming, especially, for a large- scale MANET. A commonly used approach to achieving scalability is to use hierarchical (clustered) structures. Such structures usually adopt proactive routing protocols inside the clusters and reactive routing protocols outside the clusters. They can gather the up-to-date state information with lower overhead.

No matter what methods are adopted to gather the up-to-date state information, they rely heavily on local or global broadcasting. The simplest approach to broadcasting is flooding. However, flooding is wasteful in bandwidth because of the large number of superfluous packets that are unnecessarily transmitted. Additionally, in the wireless environment, bursts of traffic resulting from massive broadcasts cause congestion resulting in collisions and even more retransmissions. Therefore, a reliable and efficient broadcast approach is necessary.

二、研究目的

The performance of QoS routing protocols relies on the availability of precise state information (e.g., bandwidth, delay and cost). However, the state information continuously changes due to load fluctuation and the dynamic nature of MANETs. Therefore, no matter what methods are adopted to gather the up-to-date state information, they rely heavily on local or global broadcasting. In fact, network layer broadcast is a key service for MANETs. It provides important control and route establishment functionality for a number of unicast and multicast protocols.

In the second year of this project, we first introduce some of previous work on MAC and network layer broadcast, and discuss some problems incurred of them. In fact, in order to optimize the performance of network layer broadcasts, the broadcast problem should be considered with the MAC and network layers. We formulate the cross-layer broadcast problem as an optimization problem and show that it is NP-hard. 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 Tones (BPBT), is proposed. Finally, BPBT is compared with previous protocols for performance evaluation by simulation.

三、文献探討

Many broadcast protocols that are devoted to the network layer have been proposed in the literature [13-25]. The primary goal of them is to reduce the total number of retransmissions needed to reach all nodes in the network. In [26], they have been categorized into four families: Simple Flooding [13], [14], Probability Based Methods [15], Area Based Methods [15], [16] and Neighbor Knowledge Methods [17-25]. Simple Flooding requires each node to rebroadcast all packets. Probability Based Methods use some basic understanding of the network topology to assign a probability to a node to rebroadcast. Area Based Methods assume nodes have common transmission distances; a node will rebroadcast only if the rebroadcast will reach sufficient additional coverage area. Neighbor Knowledge Methods maintain state on their neighborhood, via “Hello” packets, which is used in the decision to rebroadcast.

The execution of these network layer broadcast protocols relies heavily on a reliable and efficient MAC layer broadcast. The IEEE 802.11 MAC [27] utilizes an RTS/CTS/Data/ACK (Acknowledgment) procedure to account for the hidden terminal problem when unicasting packets. However, the RTS/CTS/Data/ACK procedure is too cumbersome to implement for broadcast packets as it would be difficult to coordinate and bandwidth expensive. Therefore, the only requirement made for broadcasting nodes is that they assess a clear channel before broadcasting. Unfortunately, clear channel assessment does not prevent collisions from hidden nodes. Additionally, no recourse is provided for collision when two neighbors assess a clear channel and transmit simultaneously. Consequently, the increasing probability of lost packets may deteriorate the quality of network layer broadcasts.

Some MAC layer broadcast protocols are proposed for improving the reliability of network layer broadcasts. They include BSMA [28], BMW [29], BMMM [30], LAMM [30], and others (see [31-33]). In [28], BSMA extended the IEEE 802.11 control frames: RTS, CTS and Negative Acknowledgement (NAK), to enhance the reliability. However, BSMA was not able to coordinate the transmission of CTS frames, which brought about a serious problem: the collision of CTS frames. The broadcast protocols proposed in [29-32] worked around the collision problem. Although they could avoid the collision problem, they induced another problem, the hidden terminal problem, at the same time. They are briefly described below.

Each CTS frame in unicast not only notifies a sender that the intended receiver of it is ready to receive data packets, but also prevents hidden terminals from transmission. However, in order to avoid the collision of CTS frames in broadcast, CTS frames are either transmitted at different times [30], or replaced with a CTS frame from a leader node [29], [32]. Both methods fail to prevent hidden terminal from transmission. The broadcast protocol proposed in [31] used different method to avoid the collision problem. It also fails to prevent hidden terminals from transmission.

The collision problem was solved in [33] by using a tone to signal a NAK or NCTS (Negative

CTS). In [33], when a sender has data packets to broadcast to some of its neighboring nodes, it transmits a RTS frame to inform intended receivers of the request for broadcasting. The sender broadcasts data packets if no NCTS tone is sensed and assumes that the broadcast transmission is successful if no NAK tone is sensed. However, the NAK and NCTS tones fail to guarantee the reliability of broadcast because some intended receivers that are in the coverage of other sending nodes cannot receive the RTS frame. They are unaware of their responsibility for triggering the rebroadcast or stopping the sender from broadcasting (i.e., they will not transmit a NAK tone or a NCTS tone even if they cannot receive the data packets).

On the other hand, the broadcast protocols in [28-32] may suffer from another serious problem as explained as follows. In order to enhance the reliability of broadcasts, retransmission will succeed only when all intended receivers have received the data packets. In previous protocols, a sender will broadcast data packet if it has received a CTS frame from an intended receiver. Collision of data packets may happen frequently because other intended receivers may not be ready to receive the data packet. Such collisions result in excessive retransmission of data packets.

四、研究方法

一. MAC layer Broadcast

When a node u is required to send data packets to its neighboring nodes, each neighboring node v of u has three possible states: *receiving*, *ready* and *non-ready*. They represent that v is “receiving”, “ready to receive”, and “not ready to receive” data packets, respectively. Two examples of *non-ready* v are when v is transmitting data packets (i.e., v cannot receive data packets) and when v is in the coverage of other sending node but v is not one of its intended receivers.

The broadcast problem at the MAC layer is to guarantee that all intended receivers of u have received the data packets successfully. There are two challenges of the problem. The first is to obtain the current states of neighboring nodes. The second is to decide whether u can transmit data packets or not according to the states. In this section, we only take the second challenge into account.

(一) Broadcast for Periodical Beacons

There are a number of broadcasts [34], [35] and multicasts [36], [37] at the network layer that require efficient and reliable broadcasts at the MAC layer. They include, for example, periodical beacons, alarm signals to all nodes, route discovery in on-demand routing, and multicasting video streams. In order to optimize their performances, they should be considered with the MAC and network layers. In [38], considering alarm signals to all nodes the network layer broadcast, the problem of designing a cross-layer broadcast scheme to minimize the bandwidth consumption or time delay was proved NP-hard.

In this sub-section, another network layer broadcast, i.e., periodical beacons, is considered. In many routing and multicasting protocols (e.g., see [39], [40]), beacons were emitted periodically to update routing (or multicasting) tables. Suppose that there is a set of nodes in a MANET which

are required to send beacons to their neighbors. We intend to develop a cross-layer approach to minimize the completion time. The MANET 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 is further explained as follows.

Let $V' \subset V$ be the set of nodes that are required to send beacons to their neighbors, $S=\{s_0, s_1, s_2, \dots\}$, 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 $N_G(w)$ denote the set of neighboring nodes of w in G . T is *feasible* if for every $w \in V$ and every $p \in (N_G(w) \cup \{w\}) \cap V'$, there exists a time slot s_i such that $s_i \in T(p)$ but $s_i \notin T(q)$ for all $q \in ((N_G(w) \cup \{w\}) \cap V') - \{p\}$. Such a feasible T can assure every node in V of receiving beacons from all its neighboring nodes in V' . Also we note that $T(p)$ and $T(q)$ are not a subset of each other. The problem is to determine an optimal feasible mapping, denoted by T^* , so that $\max\{i: s_i \in T^*(u)\}$ for some $u \in V'$ is minimized. Since S is infinite, T^* surely exists.

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 inquires if the vertices of G can be assigned with three distinct colors so that no two adjacent vertices have the same color [41]. In other words, we want to determine if there exists a feasible mapping F from V to $\{0, 1, 2\}$ so that $F(u) \neq F(v)$ for all $(u, v) \in E$.

Suppose that $G=(V, E)$ is an arbitrary instance of the 3-coloring problem. The following is 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 a collection of nodes that are required to broadcast beacons.

- $V_b=V \cup M$, where $m_{u,v} \in M$ if and only if $(u, v) \in E$.
- $V'_b=V$.
- $E_b=\{(u, m_{u,v}), (m_{u,v}, v): \text{for all } (u, v) \in E\}$.

Figure 1 illustrates the reduction by an example. Let $v^* = \max\{i: s_i \in T^*(u) \text{ for some } u \in V'_b\}$.

Then, $v^* < 3$ if and only if there exists a feasible F for G , as explained below.

When $v^* < 3$, $T^*(u) \in \{\{s_0\}, \{s_1\}, \{s_2\}, \{s_0, s_1\}, \{s_0, s_2\}, \{s_1, s_2\}, \{s_0, s_1, s_2\}\}$ for all $u \in V'_b$.

Since T^* is feasible, $T^*(u)$ and $T^*(v)$ are not a subset of each other for all $(u, v) \in E$. Hence, if $T^*(u)=\{s_0, s_1, s_2\}$, then u must be isolated in G (i.e., the degree of u is zero). If $T^*(u)=\{s_0, s_1\}$, then $T^*(v) \in \{\{s_2\}, \{s_0, s_2\}, \{s_1, s_2\}\}$ for all $(u, v) \in E$. The discussion is similar for other $T^*(u)$. A feasible F for G can be determined as follows. $F(u)=0$ if $T^*(u) \in \{\{s_0\}, \{s_0, s_1\}\}$, $F(u)=1$ if $T^*(u) \in \{\{s_1\}, \{s_1, s_2\}\}$, and $F(u)=2$ if $T^*(u) \in \{\{s_2\}, \{s_0, s_2\}\}$. $F(u)$ can be any if $T^*(u)=\{s_0, s_1, s_2\}$.

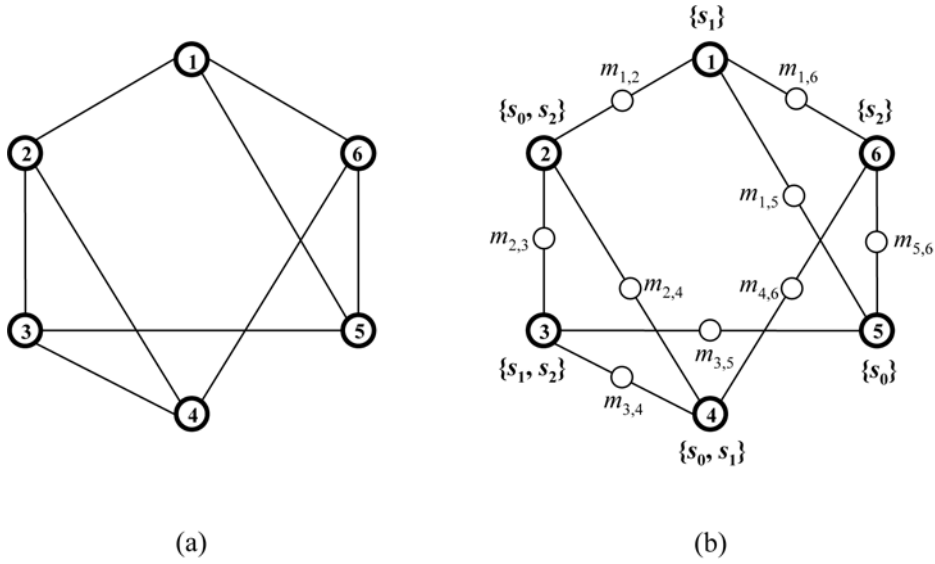


Figure 1. An example of the reduction from (a) G to (b) G_b .

For the example of Figure 1, a T^* for G_b and V'_b can be determined as follows: $T^*(1)=\{s_1\}$, $T^*(2)=\{s_0, s_2\}$, $T^*(3)=\{s_1, s_2\}$, $T^*(4)=\{s_0, s_1\}$, $T^*(5)=\{s_0\}$ and $T^*(6)=\{s_2\}$, where $v^*=2$. A feasible F for G can be determined as follows: $F(1)=1$, $F(2)=2$, $F(3)=1$, $F(4)=0$, $F(5)=0$ and $F(6)=2$. It is not difficult to see that $F(u) \neq F(v)$ for all $(u, v) \in E$.

On the other hand, suppose that $v^* \geq 3$ and there exists a feasible F for G . Another feasible mapping T for G_b and V'_b can be derived as follows: for all $u \in V$, $T(u)=\{s_i\}$ if $F(u)=i$. There is a contradiction because $v^* \geq 3$ and $\max\{i: s_i \in T(u) \text{ for some } u \in V'_b\}=2$. Q.E.D.

As a consequence of Theorem 3.1, we intend to find a feasible T so that the value of $\max\{i: s_i \in T(u) \text{ for some } u \in V'\}$ approximates the value of $\max\{i: s_i \in T^*(u) \text{ for some } u \in V'\}$. Let A_G be the adjacency matrix of G . We use G^2 to denote the graph whose adjacency matrix is $(A_G+I)^2$, where I is the identity matrix. In other words, each edge in G^2 corresponds to a path of length one or two in G . Further, we use G' to denote the induced subgraph [42] of G^2 by V' , i.e., G' is obtained from G^2 by removing those vertices that are not in V' . The meaning of G' is to relate two vertices with an edge if they may cause collision. Figure 2 depicts G^2 and G' , where G is the same as Figure 1(a) and $V'=\{2, 4, 5, 6\}$ is assumed. Notice that collision will happen if any two neighboring nodes in G' broadcast at the same time slot. The following is an approximation algorithm that can serve for our purpose.

```

t ← -1;
U ← V';
while U is not empty do
  begin
    t ← t + 1;
    for each u ∈ U do
      if T(v) ≠ {s_t} for all v ∈ N_{G'}(u) then

```

```

begin
     $T(u) \leftarrow \{s_t\};$ 
     $U \leftarrow U - \{u\};$ 
end;

end;

output( $T(u)$ ) for all  $u \in V'$ ;
output( $t$ ).

```

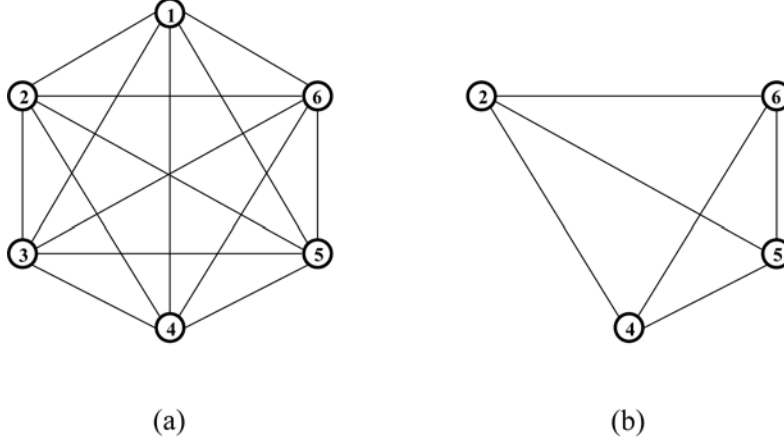


Figure 2. An example of (a) G^2 and (b) G' .

In the algorithm above, the outer while-loop considers time slots, sequentially, starting at s_0 . The inner for-loop schedules as many nodes in V' as possible at the time slot s_t so that no collision will happen. Each node $u \in V'$ can be scheduled at s_t if all of its neighboring nodes are *ready* (or s_t was not occupied by any of its neighboring nodes in G'). We note that each $T(u)$ contains only one time slot, i.e., $|T(u)|=1$ for all $u \in V'$. It is not difficult to see that the final T is feasible and the final t gives the value of $\max\{i: s_i \in T(u) \text{ for some } u \in V'\}$. For the example of Figure 2, if the nodes in V' are examined in the sequence of 2, 4, 5 and 6, then a feasible T will be generated as follows: $T(2)=\{s_0\}$, $T(4)=\{s_1\}$, $T(5)=\{s_2\}$ and $T(6)=\{s_3\}$. The value of $\max\{i: s_i \in T(u) \text{ for some } u \in V'\}$ (or the final t) is equal to 3.

Let $\Delta(G)$ be the maximum degree [42] of G , i.e., $\Delta(G)=\max\{|(N_G(w))|: \text{for all } w \in V\}$. Also define $\text{MB}(G, V')=\max\{|(N_G(w) \cup \{w\}) \cap V'|: \text{for all } w \in V\}$. An intuitive meaning of $\text{MB}(G, V')$ is explained as follows. Let us consider $N_G(w) \cup \{w\}$ the neighborhood of w , and so $|(N_G(w) \cup \{w\}) \cap V'|$ gives the number of nodes in this neighborhood that are required to broadcast. $\text{MB}(G, V')$ then takes the maximum of $|(N_G(w) \cup \{w\}) \cap V'|$ among all nodes w of V . In the following, two lemmas are introduced that suggest a lower bound on the value of $\max\{i: s_i \in T^*(u) \text{ for some } u \in V'\}$ and an upper bound on the value of $\max\{i: s_i \in T(u) \text{ for some } u \in V'\}$. By their aid, an approximation ratio can be obtained.

Lemma 1. The value of $\max\{i: s_i \in T^*(u) \text{ for some } u \in V'\}$ is bounded below by $\text{MB}(G, V') - 1$.

Proof. Suppose $w \in V$ and $p \in (N_G(w) \cup \{w\}) \cap V'$. Since T^* is feasible, there exists a time slot s_i such that $s_i \in T^*(p)$ and $s_i \notin T^*(q)$ for all $q \in ((N_G(w) \cup \{w\}) \cap V') - \{p\}$. In other words, T^* uses at

least $|(N_G(w) \cup \{w\}) \cap V'|$ time slots and hence $\max\{i: s_i \in T^*(u) \text{ for some } u \in V'\} \geq \max\{|(N_G(w) \cup \{w\}) \cap V'|: \text{for all } w \in V\} - 1 = \text{MB}(G, V') - 1$. Q.E.D.

Lemma 2. Suppose that T is the feasible mapping generated by the approximation algorithm. The value of $\max\{i: s_i \in T(u) \text{ for some } u \in V'\}$ is bounded above by $\Delta(G')$.

Proof. Suppose conversely that $T(v) = s_{\Delta(G')+1}$ for some $v \in V'$. Since $|N_{G'}(v)| \leq \Delta(G')$, according to the pigeonhole principle [43], a time slot in $\{s_0, s_1, \dots, s_{\Delta(G')}\}$ can be found that was not occupied by $N_{G'}(v)$. So, v should be scheduled before $s_{\Delta(G')+1}$ by the algorithm, which is a contradiction.

Q.E.D.

The *approximation ratio* [44] of an approximation algorithm is the ratio of the approximate value obtained by the algorithm to the optimal value. The approximation ratio of the proposed algorithm is computed as $\frac{\max\{i: s_i \in T(u) \text{ for some } u \in V'\}}{\max\{i: s_i \in T^*(u) \text{ for some } u \in V'\}}$, which is not greater than

$$\frac{\Delta(G')}{\text{MB}(G, V') - 1} \text{ by Lemma 1 and Lemma 2.}$$

THEOREM 2. The approximation ratio of the proposed algorithm is bounded above by $\frac{\Delta(G')}{\text{MB}(G, V') - 1}$.

(二) A General MAC Layer Broadcast Scheme

In last section, a particular network layer broadcast, namely periodical beacons, was considered, and an approximation algorithm for optimal broadcast was proposed. Recall that the broadcast problem at the MAC layer is to decide whether a sender can transmit data packets or not. In the proposed algorithm of last section, a sender will transmit data packets if all of its neighboring nodes are *ready*. It implies a MAC layer broadcast scheme for periodical beacons.

However, periodical beacon is the only one of many network layer broadcasts (multicasts). In this sub-section, we propose a general MAC layer broadcast scheme to satisfy more network layer broadcasts (multicasts). A moment of an active MANET can be considered that some nodes want to unicast or broadcast packets to some of their neighbors. When the general network layer broadcast (multicast) is considered, the broadcast problem can be generalized as follows.

Suppose that $V' = \{v_0, v_1, v_2, \dots, v_{|V'|-1}\}$ is the set of nodes that are required to send data packets to some or all of their neighbors. For $0 \leq j \leq |V'| - 1$, let D_j denote the set of neighboring nodes of v_j that have to receive data packets from v_j , where $1 \leq |D_j| \leq |N_G(v_j)|$. A feasible mapping T from V' to 2^S requires that for every $w \in V$ and every $v_x \in N_G(w) \cup \{w\}$ with $w \in D_x$, there exists a time slot s_i such that $s_i \in T(v_x)$ but $s_i \notin T(v_y)$ for all $v_y \in (N_G(w) \cup \{w\}) - \{v_x\}$. An optimal feasible mapping T^* is to minimize $\max\{i: s_i \in T^*(v_j) \text{ for some } v_j \in V'\}$. The broadcast problem for periodical beacons is an instance of the general broadcast problem in which $D_j = N_G(v_j)$ for all $0 \leq j \leq |V'| - 1$.

We note that the problem of finding a T^* for the general broadcast problem is NP-hard. The approximation algorithm proposed in last section can be applied to generate a feasible T for the

general broadcast problem. Besides, no collision will happen. Surprisingly, moderate collision may reduce the completion time of the general broadcast problem. Refer to Figure 3, where $V=\{v_0, v_1, v_2, 1, 2, 3, 4, 5, 6, 7\}$, $V'=\{v_0, v_1, v_2\}$, $D_0=\{1, 2, 5\}$, $D_1=\{3, 4, 7\}$ and $D_2=\{3, 4, 6\}$. If the approximation algorithm of Section 3.2.1 is applied, a feasible T with $T(v_0)=\{s_0\}$, $T(v_1)=\{s_1\}$ and $T(v_2)=\{s_2\}$ will be obtained which has $\max\{i: s_i \in T(v_j) \text{ for some } v_j \in V'\}=2$. However, there exists another feasible T' with $T'(v_0)=\{s_0, s_1\}$, $T'(v_1)=\{s_0\}$ and $T'(v_2)=\{s_1\}$, which causes $\max\{i: s_i \in T'(v_j) \text{ for some } v_j \in V'\}=1$. Although T' causes collision, it reduces the completion time. It is because moderate collision may enhance the spatial reuse of bandwidth. In the algorithm proposed in Section 3.2.1, although no collision will happen, the spatial reuse of bandwidth is also degraded.

When a sender broadcasts data packets to intended receivers, collision will happen at *receiving* and *non-ready* nodes. Collision at a *receiving* node costs highly because it brings about more retransmissions. In order to avoid such inefficiency, the sender had better deferred its data transmission if one of its intended receivers is *receiving*. With this thought, we design the following algorithm to generate a feasible T for the general broadcast problem. The algorithm is indeed a modification of the previous one in Section 3.2.1. Unlike the former T , the newly generated T may induce collision. Besides, each $T(v_j)$ may contain more than one time slot.

```

t ← -1;
set T(vj) empty for all vj ∈ V';
U ← V';
while U is not empty do
  begin
    t ← t + 1;
    for each vj ∈ U do
      if each vx ∈ NG(vj) is not receiving and there exists
        one ready vy ∈ Dj then
        begin
          T(vj) ← T(vj) + {st};
          Dj ← Dj - {vr: vr ∈ Dj and vr is ready};
          if Dj is empty then U ← U - {vj};
        end;
    end;
  output(T(vj)) for all vj ∈ V';
output(t).

```

Similarly, the outer while-loop considers time slots sequentially, and the inner for-loop schedules as many nodes in V' as possible at the current time slot. A node $v_j \in V'$ can be scheduled at the current time slot if none of its neighboring nodes is *receiving* and one or more of its intended receivers are *ready*. A node $v_x \in N_G(v_j)$ is *receiving* at s_t if there exists v_z so that $s_t \in T(v_z)$, $v_x \in D_z$ and $v_x \notin N_G(v_p)$ for all $v_p \neq v_z$ with $s_t \in T(v_p)$. A node $v_y \in D_j$ is *ready* at s_t if $v_y \notin N_G(v_q)$ for all $v_q \neq v_j$ with $s_t \in T(v_q)$. The final T is feasible and the final t gives the value of $\max\{i: s_i \in$

$T(v_j)$ for some $v_j \in V'$. For the example of Figure 3, a feasible T with $T(v_0)=\{s_0, s_1\}$, $T(v_1)=\{s_0\}$ and $T(v_2)=\{s_1\}$ will be generated which has $\max\{i: s_i \in T(v_j) \text{ for some } v_j \in V'\}=1$.

二. A MAC Layer Broadcast Protocol with Busy Tone

In the proposed algorithm of last section, a sender will transmit data packets if none of its neighboring nodes is *receiving* and one or more of its intended receivers are *ready*. It implies a MAC layer broadcast scheme for the general network layer broadcast. Based on this scheme, an efficient and reliable MAC layer broadcast protocol, named BPBT, is then proposed in this section. In BPBT, a narrow-bandwidth busy tone is implemented with enough spectral separation on the single shared channel. We use the busy tone to acknowledge RTS frames and data packets, and provide continuous protection for the transmitted data packets. All nodes sensing any busy tone are not allowed to send RTS requests. In fact, the busy tone approach was used before to avoid the hidden terminal problem in unicast (see [47], [45], [46]).

Two reasons of BPBT applying busy tone are described as follows. Recall that the algorithm of last section requires a sender to examine the current states of its neighboring nodes before transmission. When the RTS/CTS handshake is only applied, the sender will be unaware of the state of an intended receiver if it does not receive a CTS from the receiver. The receiver may not receive the RTS frame because it is in the coverage of other sending nodes, or the receiver cannot reply CTS because it is in the coverage of other receiving nodes. So, BPBT uses RTS frames and busy tone to help the sender detect the current states of its neighboring nodes. In consequence of examining the current states of neighboring nodes and adopting the efficient MAC layer broadcast scheme, the hidden terminal problem is solved and the spatial reuse of bandwidth is improved. Another reason of BPBT applying busy tone is to enhancing the reliability of broadcast. In BPBT, the busy tone is used to acknowledge the data packets and RTS frames instead of the ACK and CTS frames, respectively. It is because the ACK and CTS frames may collide with data packets received at other receivers.

Each node is equipped with two antennas; one is for busy tone and the other is for data transmission. When a node v_i has data packets to broadcast to some of its neighboring nodes, it senses the busy tone channel first and transmits the RTS frame if the busy tone channel is free (which means that no one in its transmission area is *receiving*). Assume that the set of intended receivers of v_i is $D_i=\{u_1, u_2, \dots, u_k\}$. RTS frame not only informs intended receivers of the request

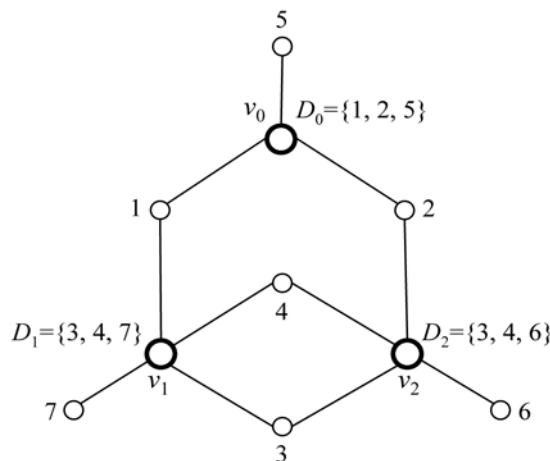


Figure 3. An instance for the general broadcast problem.

for broadcasting, but also assigns a waiting time WT_j to u_j , for all intended receiver $u_j \in D_i$. The waiting times of all intended receivers are different to each other. The use of the waiting time is to schedule the acknowledgements of data packets.

When an intended receiver receives the RTS frame, it turns on its busy tone signal, replying to the sender and announcing that it is waiting for the incoming data packet. When the sender senses the busy tone signal (which means that one or more of its intended receivers are *ready*), it knows that its broadcast request has been successful. It waits a mandatory waiting time $t_{mw}=2\tau$, where τ is the maximum propagation delay between two neighboring nodes [45]. After the mandatory waiting time expiring, the sender broadcasts the data packets. The mandatory waiting time is meant to allow all possible RTS transmissions in the range of an intended receiver to be aborted. It can avoid RTS packets sent from some hidden node to collide with data packets.

We give an extreme example to explain the use of the mandatory waiting time. Assume that the propagation delay between a sender v_i and an intended receiver u_j is zero, and the propagation delay between u_j and a hidden node w from v_i is τ . When u_j receives the RTS frame correctly at time t_0 , u_j turns on its busy tone signal at time t_0 . Since the propagation delay between v_i and u_j is zero, v_i can sense the busy tone signal at time t_0 and will broadcast the data packets at time t_0+t_{mw} . So the time of data packets arriving at u_j is t_0+t_{mw} . On the other hand, the hidden node w senses the busy tone signal at time $t_0+\tau$ because the propagation delay between w and u_j is τ . Then w aborts the active RTS transmission at time $t_0+\tau$ and the data channel at u_j is busy until time $t_0+2\tau$. Therefore, in order to avoid the aborted RTS to collide with data packets received at u_j , t_{mw} should be assigned to 2τ .

By the end of data transmission, the sender continuously monitors the busy tone channel in a time period WT_{ack} for the acknowledgements replied from its intended receivers. Upon successful reception of the data packet, for all intended receiver $u_j \in D_i$, u_j turns off the busy tone signal and waits for a time period WT_j . After the time expiring, then u_j turns on the busy tone during a short time interval t_a to acknowledge the broadcasts data frames. In order to provide continuous protection for the acknowledgements, the sender continuously transmits the noise packets during the time period WT_{ack} . The noise packets can avoid some nodes, that are neighboring to v_i and not belong to D_i , to receive a RTS frame from other senders and turn on their busy tone. Figure 4 shows the time diagram of BPBT.

Considering the propagation delays between the sender and all of its intended receivers are different. The value of WT_{ack} and WT_j for all $u_j \in D_i$ should be calculated carefully. Assume that the sender v_i ends the data transmission at time t_0 . The latest time of busy tone signals for protecting data packets arriving at v_i may be $t_0+2\tau$. So the acknowledgements of data packets should be replied during the time interval from $t_0+2\tau$ to t_0+WT_{ack} . In order to schedule the acknowledgements, the time interval from $t_0+2\tau$ to t_0+WT_{ack} can be divided into k equal parts. The sender can wait each acknowledgement in each part. Assume that the acknowledgement of the intended receiver u_1 should be arrived at the first part. If the propagation delay between u_1 and v_i is zero, the time of the acknowledgement arriving at v_i is t_0+WT_1 . If the propagation delay between u_1 and v_i is τ , the time of the acknowledgement arriving at v_i is $t_0+2\tau+WT_1$. Therefore, the first part should be $[t_0+WT_1, t_0+2\tau+WT_1+t_a]$. Since the beginning of the first part is $t_0+2\tau$, the first part is

$[t_0+2\tau, t_0+2\tau+2\tau+t_a]$. The rest may be deduced by analogy and the detail is shown in Figure 4.

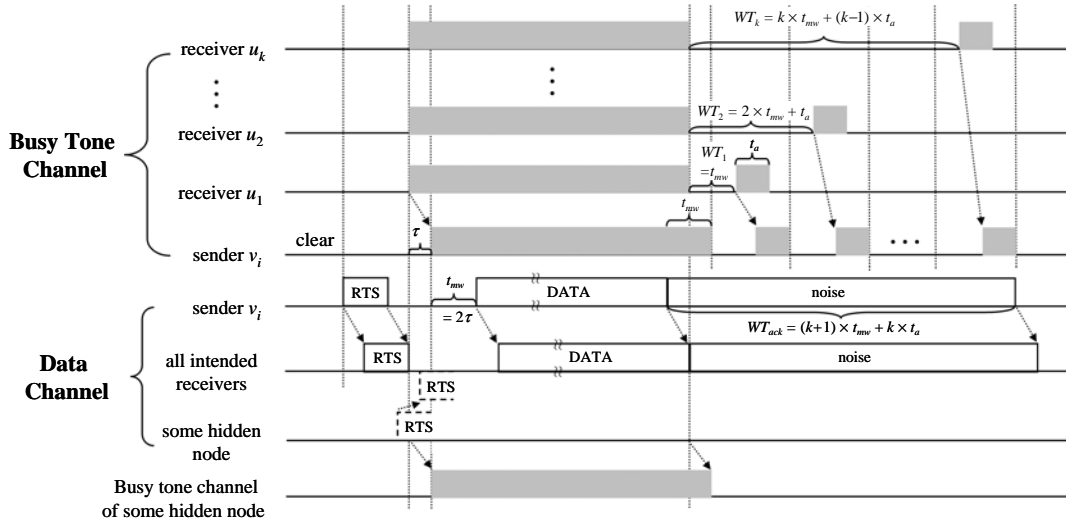


Figure 4. Time diagram of BPBT.

五、結果與討論

Simulation was made to evaluate the performance of BPBT by comparing BPBT with BMW and BMMM for three network layer broadcasts, i.e., periodical beacons, alarm signals to all nodes and multicasting video streams. Simulation modules were developed by C++ to simulate the behavior of each protocol. We simply describe the behavior of BMW and BMMM as follows. The basic idea of the BMW protocol is to treat each broadcast request as multiple unicast requests. In BMW, when a node has a broadcast packet to send, it performs RTS/CTS handshake for an intended receiver. If RTS/CTS handshake is successful, the node transmits all missing data packets (which failed to be received at the receiver in previous broadcasts) to the receiver and waits for an ACK.

Other intended receivers may also receive the data packets successfully at the same time. However, their acknowledgements are deferred until the node performs RTS/CTS handshake for them. If the sender does not perform RTS/CTS handshake for them immediately, some intended receivers that have not received the data packets may receive stale and useless packets later. Hence, in the implementation of BMW in our simulation, the sender is triggered to rebroadcast the data packet immediately until it has received the ACKs from all intended receivers. On the other hand, in BMMM, when a node has a broadcast packet to send, it performs RTS/CTS handshake for each intended receiver one by one to obtain their current states. Then it will broadcast the data packet if it has received a CTS frame from an intended receiver. By the end of its data transmission, the sender also performs RAK (Request for ACK) /ACK handshake for each intended receiver one by one.

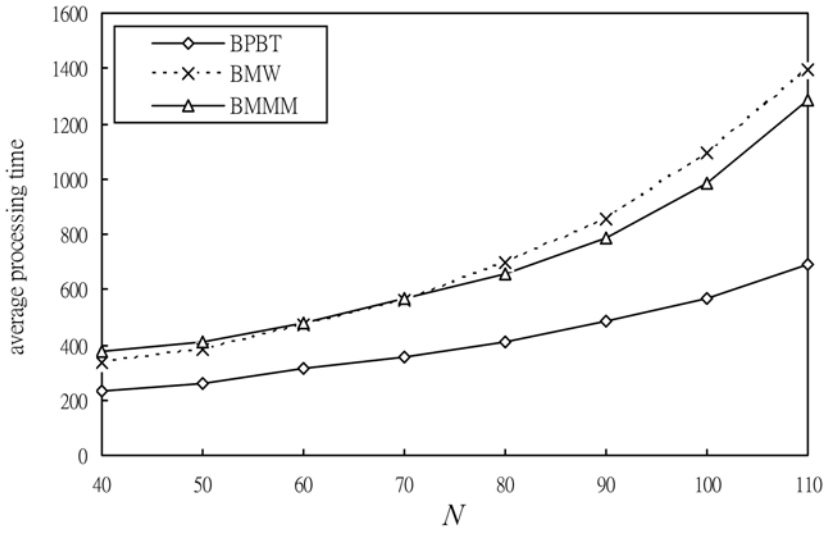


Figure 5. Average processing times of periodical beacons with different values of N .

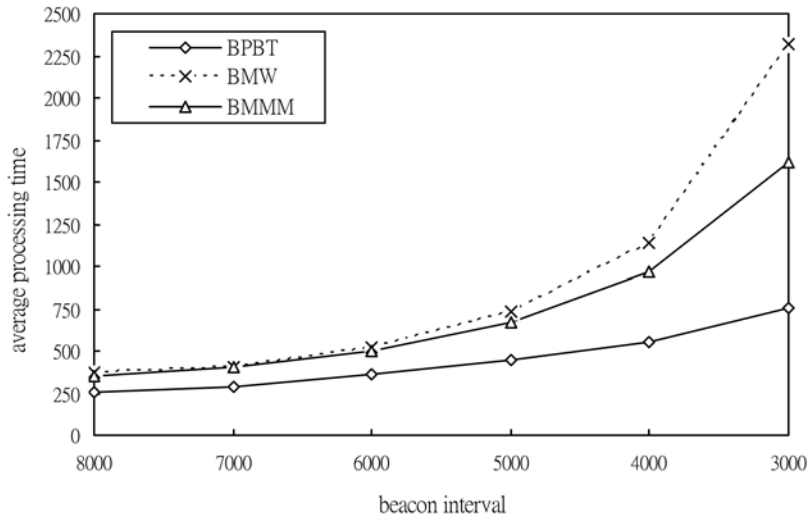


Figure 6. Average processing times of periodical beacons with different beacon intervals.

In our simulation experiments, a MANET of N nodes which were randomly spread in a $400\text{m} \times 400\text{m}$ area was simulated, where N is a parameter. Nodes were free to move anywhere within this area. The radio transmission range of each 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 100 time slots (5 time slots). We assume that τ and t_a are 1 time slot. The simulation continued for 50000000 time slots. The mobility model was as follows. First 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 50000 time slots. If it decided to rest, it would randomly choose a rest period from 3000000 to 15000000 time slots. When it finally reached the spot or exhausted the rest period, it would decide to migrate or rest again.

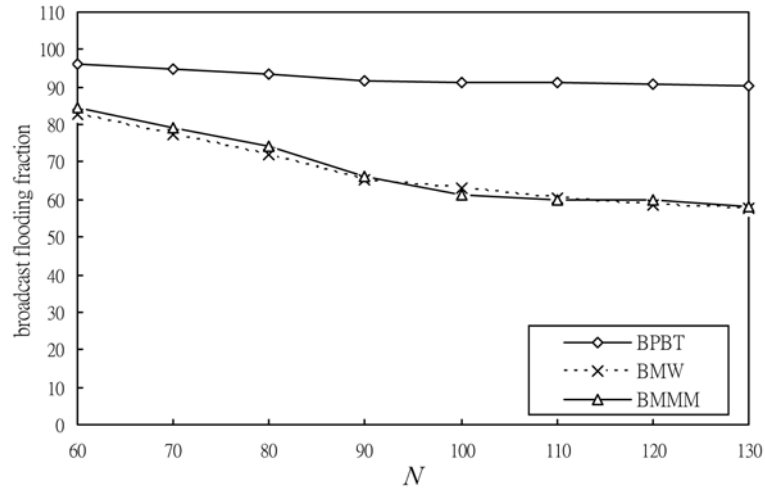


Figure 7. Broadcast flooding fractions of alarm signals to all nodes with different values of N .

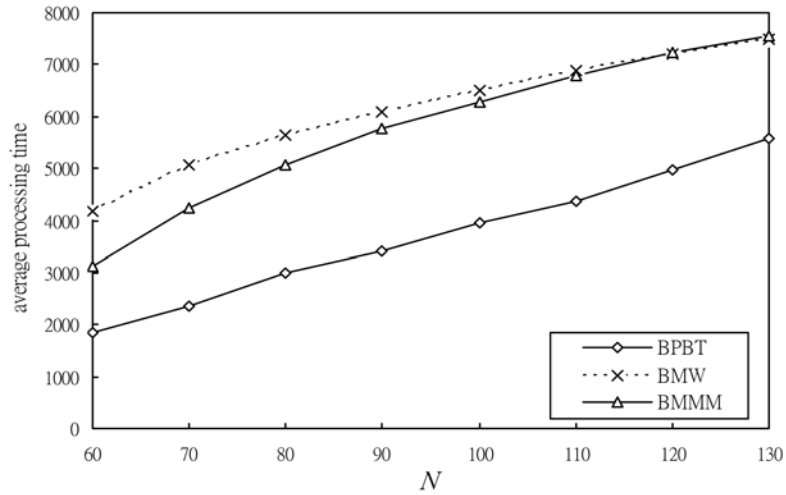


Figure 8. Average processing times of alarm signals to all nodes with different values of N .

(一) Periodical Beacons

All nodes broadcast “Hello” packets (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. Figure 5 and Figure 6 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 Figure 5 was assumed 10000 time slots and $N=50$ was assumed in Figure 6. As observed from Figure 5 and Figure 6, 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.

The RTS/CTS handshakes in BMMM seem likely to be able to solve the hidden terminal

problem because the MANET is almost static for the average processing time in our mobility model. However, the consuming time to carry the cumbersome handshaking procedure out is long. The states of neighboring nodes may change due to other broadcasting transmissions. The stale states result in a wrong decision of transmission. The performance of BMW is the worst one of all due to serious collisions. This is because only one neighboring node would be demanded to reply CTS. The hidden terminal problem in BMW is more serious than in BMMM.

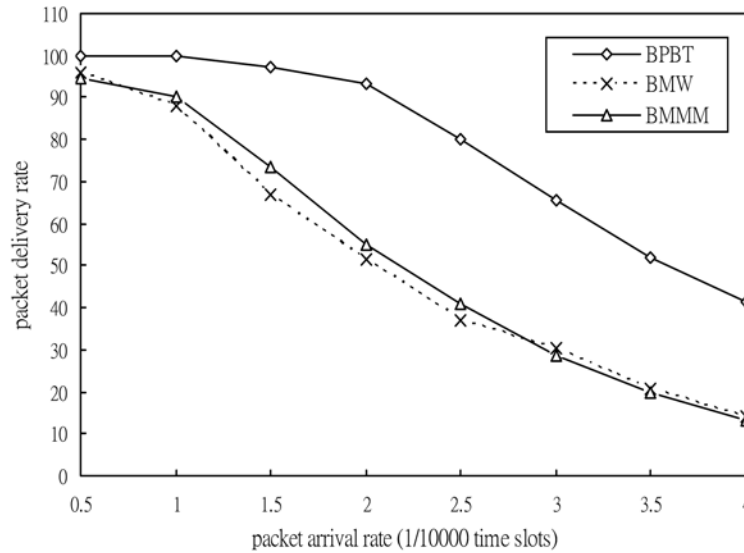


Figure 9. Packet delivery rates of multicasting video streams with different packet arrival rates.

(二) Alarm Signals to All Nodes

Some nodes were asked to broadcast packets to all nodes by flooding. The broadcast frequency was assumed once per 10000 time slots. That is, one node was randomly selected to broadcast every 10000 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 packets. 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. Since the transmission range of each node and the area of the MANET are constant, the diameter of the MANET is constant and would not be the dominating factor of the average processing time. Figure 7 and Figure 8 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.

(三) Multicasting Video Streams

There were three multicast groups whose source nodes and multicast members were not changed throughout the simulation. Each source node was asked to transmit multicast packets to

the nodes of the same multicast group. Since the traffic was real-time video streams, each multicast packet carried a delay threshold by which a node having the multicast packet could decide to discard it or transmit it further. Figure 9 showed the packet delivery rates of BPBT, BMW and BMMM with different packet arrival rates. BPBT has higher packet delivery rates than BMW and BMMM because the reliability of data transmission and the spatial reuse of bandwidth are excellent in BPBT.

It can be observed from Figure 9 that the packet delivery rates decrease fast when the packet arrival rates exceed some thresholds: they are 2 for BPBT and 1 for BMW and BMMM. The thresholds mainly depend on three factors: network bandwidths, delay thresholds and performances of protocols. The first two factors were constant for the experiment of Figure 9. Figure 10 further showed the packet delivery rates of BPBT when the packet arrival rates varied and the delay thresholds were 10000, 20000, 30000 and 40000 time slots.

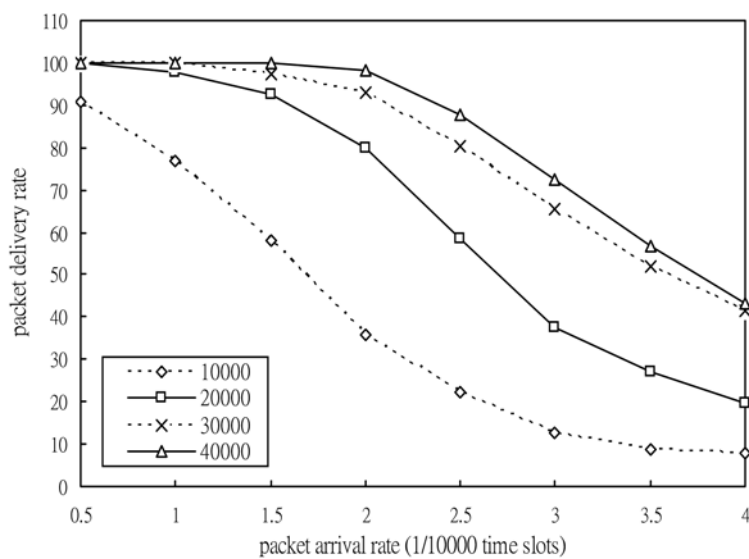


Figure 10. Packet delivery rates of BPBT with different packet arrival rates and different delay thresholds.

六、参考文献

- [1] J. Z. Sun, "Mobile ad hoc networking: an essential technology for pervasive computing," *Proceedings of the IEEE International Conference on Info-tech and Info-net (ICII)*, pp. 316-321, October 2001.
- [2] L. Georgiadis, P. Jacquet, and B. Mans, "Bandwidth reservation in multihop wireless networks: complexity, mechanisms," *Proceedings of the IEEE International Conference on Distributed Computing Systems Workshops (ICDCSW'04)*, pp. 762-767, March 2004.
- [3] C. R. Lin and J. S. Liu, "QoS routing in ad hoc wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 1426-1438, August 1999.
- [4] S. Chen and K. Nahrstedt, "Distributed quality-of-service routing in ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 1488-1505, August 1999.

- [5] C. Zhu and M. S. Corson, "Bandwidth calculation in a TDMA-based ad hoc network," *ISR Technical Reports*, CSHCN 2000-47, 2000.
- [6] W. H. Liao, Y. C. Tseng, and K. P. Shih, "A TDMA-based bandwidth reservation protocol for QoS routing in a wireless mobile ad hoc network," *IEEE International Conference on Communications (ICC)*, pp. 3186-3190, May 2002.
- [7] Y. S. Chen, Y. C. Tseng, J. P. Sheu, and P. H. Kuo, "An on-demand, link-state, multi-path QoS routing in a wireless mobile ad-hoc network," *Computer Communications*, vol. 27, pp. 27-40, January 2004.
- [8] Q. Xue and A. Ganz, "Ad hoc QoS on-demand routing (AQOR) in mobile ad hoc networks," *Journal of Parallel and Distributed Computing*, vol. 63, pp. 154-165, February 2003.
- [9] C. H. Hsu, Y. L. Kuo, H. K. Wu, and G. H. Chen, "QoS routing in mobile ad hoc networks based on the enhanced distributed coordination function," *Proceedings of the IEEE Vehicular Technology Conference (VTC)*, pp. 2663-2667, September, 2004.
- [10] R. Sivakumar, P. Sinha, and V. Bharghavan, "CEDAR: a core-extraction distributed ad hoc routing algorithm," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 1454-1465, August 1999.
- [11] J. Broch, D. A. Maltz, D. B. Johnson, Y. C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing approach," *Proceedings of the ACM/IEEE international conference on Mobile computing and networking (MOBICOM)*, pp. 85-97, October 1998.
- [12] S. R. Das, C. E. Perkins, and E. M. Royer, "Performance comparison of two on-demand routing protocols for ad hoc networks," *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, pp. 26-30, March 2000.
- [13] C. Ho, K. Obraczka, G. Tsudik, and K. Viswanath, "Flooding for reliable multicast in multi-hop ad hoc networks," *Proceedings of the International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communication (DIALM)*, pp. 64-71, 1999.
- [14] J. Jetcheva, Y. Hu, D. Maltz, and D. Johnson, "A simple protocol for multicast and broadcast in mobile ad hoc networks," Internet Draft: draft-ietf-manet-simple-mbcast-01.txt, July 2001.
- [15] Y. C. Tseng, S. Y. Ni, Y. S. Chen, and J. P. Sheu, "The broadcast storm problem in a mobile ad hoc network," *ACM/Baltzer Wireless Network*, vol. 8, no. 2/3, pp. 153-167, March, 2002.
- [16] M. T. Sun, W. C. Feng, and T. H. Lai, "Location aided broadcast in wireless ad hoc networks," *Proceedings of the IEEE Conference on Global Telecommunications (GLOBECOM)*, pp. 2842-2846, 2001.
- [17] 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.
- [18] W. Peng and X. Lu, "On the reduction of broadcast redundancy in mobile ad hoc

- networks,” *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC)*, pp. 129-130, 2000.
- [19] A. Qayyum, L. Viennot, and A. Laouiti, “Multipoint relaying: An efficient technique for flooding in mobile wireless networks,” Technical Report 3898, INRIA - Rapport de recherche, 2000.
- [20] W. Peng and X. Lu, “AHBP: An efficient broadcast protocol for mobile ad hoc networks,” *Journal of Computer Science and Technology*, vol. 16, no. 2, pp. 114-125, Mar 2001.
- [21] W. Peng and X. Lu, “Efficient broadcast in mobile ad hoc networks using connected dominating sets,” *Journal of Software*, vol. 12, no. 4, pp. 529-536, 2001.
- [22] J. Sucec and I. Marsic, “An efficient distributed network-wide broadcast algorithm for mobile ad hoc networks,” CAIP Technical Report 248, Rutgers University, September 2000.
- [23] I. Stojmenovic, M. Seddigh, and J. Zunic, “Dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 13, no. 1, pp. 14-25, January 2002.
- [24] J. Wu and F. Dai, “Broadcasting in ad hoc networks based on self-pruning,” *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, pp.2240-2250, March 2003.
- [25] W. Lou and J. Wu, “On reducing broadcast redundancy in ad-hoc wireless networks,” *IEEE Transactions on Mobile Computing*, vol. 1, no. 2, pp. 111- 123, 2002.
- [26] B. Williams and T. Camp, “Comparison of broadcasting techniques for mobile ad hoc networks,” *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC)*, pp, 194-205, 2002.
- [27] IEEE 802.11 Working group, “Wireless LAN medium access control (MAC) and physical layer (PHY) specifications,” *IEEE 802.11 Standard*, September, 1999.
- [28] 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.
- [29] 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.
- [30] 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.
- [31] 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.
- [32] 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.
- [33] 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.

- [34] 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.
- [35] 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.
- [36] 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.
- [37] 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.
- [38] 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.
- [39] 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.
- [40] 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.
- [41] C. H. Padadimitriou, *Computational Complexity*, Wesley, 1995.
- [42] D. B. West, *Introduction to Graph Theory*, Prentice Hall, second edition, 2001.
- [43] R. P. Grimaldi, *Discrete and Combinatorial Mathematics*, Wesley, fifth edition, 2004.
- [44] V. V. Vazirani, *Approximation Algorithms*, Springer, 2001.
- [45] 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.
- [46] 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.
- [47] 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.

七、計畫結果自評

Network layer broadcasts, such as periodical beacons, alarm signals to all nodes, route discovery in on-demand routing, and multicasting video streams, are key services for MANETs. They provide important control and route establishment functionality for a number of unicast and multicast protocols. In order to optimize the performances of network layer broadcasts, they

should be considered with the MAC and network layers. The execution of these network layer broadcasts relies heavily on a reliable and efficient MAC layer broadcast.

One of the objectives of MAC protocols is to synchronize accesses of multiple nodes to the common channel. It is a challenge to MANETs because hidden terminals are unaware of some on-going communications. The hidden terminal problem in broadcast is more serious than in unicast. The clear channel assessment, which has been adopted in the IEEE 802.11 for broadcast, is incapable to prevent the hidden terminal problem. Additionally, no recourse is provided for collision. The increasing probability of lost packets may deteriorate the quality of network layer broadcasts.

The broadcast problem at the MAC layer is to guarantee that all intended receivers of a sender have received the data packets successfully. The two challenges of the problem are to obtain the current states of neighboring nodes and decide whether the sender can transmit data packets or not according to the states. Previous MAC layer broadcast protocols, such as BMW and BMMM, extended the IEEE 802.11 control frames (RTS, CTS and ACK) to enhance the reliability of broadcast. However, the RTS/CTS handshake is unable to examine the current states of neighboring nodes. They still suffer from the hidden terminal problem at the same time. On the other hand, the MAC layer broadcast scheme adopted in BMW and BMMM is also inefficient. A sender was permitted to start transmission as long as one of its intended receivers was *ready*. As a side effect, the broadcast scheme incurred serious collisions and excessive retransmission of data frames.

In the second year of this project, we formulate the cross-layer broadcast problem as an optimization problem and show that it is NP-hard. An approximation algorithm with a guaranteed approximation ratio is also suggested. The algorithm implies a MAC layer broadcast scheme for the general network layer broadcast. Based on this scheme, a reliable and efficient MAC layer broadcast protocol, named BPBT, is proposed. BPBT uses RTS frames and busy tone to help the sender detect the current states of its neighboring nodes. In consequence of examining the current states of neighboring nodes and adopting the efficient MAC layer broadcast scheme, the hidden terminal problem is solved and the spatial reuse of bandwidth is improved. Finally, 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, broadcast flooding fraction and packet delivery rate.