An efficient reliable one-hop broadcast in mobile ad hoc networks

Seungjin Park a, Seong-Moo Yoo b, * 

a Department of Management and Information Sciences, University of Southern Indiana, Evansville, IN 47712, USA 
b Electrical and Computer Engineering Department, The University of Alabama in Huntsville, Huntsville, AL 35899, USA

ARTICLE INFO

Article history:
Received 12 February 2011 
Accepted 29 March 2012 
Available online 16 April 2012

Keywords:
One-hop broadcast 
Mobile ad hoc network 
MAC protocol

ABSTRACT

A reliable one-hop broadcast is a fundamental communication primitive in mobile ad hoc networks in which a message from the source node is guaranteed to be delivered to all nodes within the source node's transmission range. Despite the importance of it, reliable one-hop broadcast is not easy to accomplish due to collisions in wireless networks known as Hidden Terminal Problem. This paper presents a MAC protocol that not only guarantees reliable one-hop broadcast but also achieves it efficiently by exploring as many simultaneous executions of the communication as possible. In addition to the data packets, the proposed algorithm utilizes the control packets that prevent packet collisions, and at the same time, make the simultaneous communications possible to improve the network throughput. Simulation results show the effectiveness of the proposed algorithm.

© 2012 Elsevier B.V. All rights reserved.

1. Introduction

A Mobile Ad Hoc Network (MANET) consists of a set of wireless mobile hosts (or nodes) that are free to move in any directions at any speed. A MANET does not require any preexisting fixed infrastructures, and therefore it can be built on the fly. Since nodes in a MANET operate on batteries and have limited transmission ranges, minimizing unnecessary communications is essential to improve the network life and throughput.

Due to the nature of wireless networks, a transmission in the networks is basically a one-hop broadcast, in which a signal transmitted from a node (source node) reaches all nodes within its transmission range (neighbors of the source node). Many important algorithms in MANETs heavily depend on the performance of one-hop broadcasting [11–15]. Some significant applications of one-hop broadcasting are:

(1) Reactive routing algorithms in MANETs. In most reactive algorithms in MANETs [21–24], if the source node does not have route information to the destination node, the source node floods the network with a special control packet, Route Request Packet (REQ), to discover a path to the destination. This flooding process is based on one-hop broadcast and continues until the REQ reaches the destination.

(2) Proactive routing algorithms that are based on Distance Vector Routing or Link State Routing [16–20]. In implementing a proactive algorithm, each node maintains routing information in its routing table through periodic exchanges of its information with its neighbors. This exchange is usually done in the form of one-hop broadcast.

(3) Topology control. By exchanging necessary information with its neighbors, a wireless node adjusts its transmission range so that an efficient network topology is maintained for better energy conservation and network throughput [25–27].

(4) Algorithms based on hierarchical organization of nodes. Especially, cluster-based algorithm needs one-hop broadcasting when the nodes are in the process of forming clusters. After a cluster is formed,
the head of the cluster must broadcast its election as a cluster head to all nodes in its cluster using one-hop broadcast as well [28–30].

The algorithms described above work correctly provided that the one-hop broadcasting is reliable, i.e., packet delivery from the source node to all its neighbor nodes is guaranteed. Otherwise, inaccurate and/or insufficient information may cause severe degradation of the algorithms and therefore result in undesirable results. Although many algorithms have been developed that require reliable one-hop broadcasting, thus far not many reliable one-hop broadcasting protocols actually have been proposed.

Achieving a reliable one-hop broadcast in wireless networks is not an easy task due to the collisions caused by a phenomenon known as Hidden Terminal Problem [1]. If a transmission is just between two nodes (i.e., point-to-point communication), then RTS–CTS protocol can resolve the problem eventually [16]. However, point-to-point approach will be extremely inefficient and often useless if it is used for one-hop broadcast in MANETs. Another degradation of network throughput occurs when nodes cannot explore possible simultaneous transmissions. An example of this type is the Exposed Terminal Problem [2] that prevents two neighbor nodes from successful simultaneous transmissions.

Many communication algorithms proposed thus far are based on the single-channel MAC protocols. However, IEEE 802.11 standard for wireless LAN provides multiple channels for communication between nodes in the network [1]. One of the benefits of having multiple channels is that multiple transmissions on different channels can take place simultaneously without causing any collision and contention. Therefore, higher network throughput can be achieved if the multiple channels are used carefully [2–7].

This paper presents a reliable one-hop broadcasting algorithm, Efficient Reliable One-Hop Broadcasting (EROB), that guarantees the delivery of broadcast message from the transmitting node to all nodes within its transmission range. EROB uses three different channels, one for data packets transmissions and two for control packet transmissions to avoid the collision. EROB uses different transmission ranges for control and data transmission for further increase in the network throughput by allowing as many simultaneously transmissions as possible. Simulation results show that EROB outperforms naïve algorithm that does not provide any precaution on collisions and another algorithm that implements CSMA/CA [31].

The rest of the paper is organized as follows. Section 2 explains terminology and related works for the paper. New one-hop broadcasting algorithm is presented in Section 3 followed by simulation result in Section 4. Finally, conclusion and discussion will be presented in Section 5.

2. Preliminary

This section explains the terminology and related works that help understanding this paper.

2.1. Terminology

When a node S in a wireless network transmits a signal, the signal spreads all directions to the distance usually proportional to the transmission power. The area covered by the signal is approximated by a circle centered at S and is called transmission range (or broadcast area) and is denoted as TRS. TRS is also used to denote the radius of the area, if there is no possibility of confusion. In this paper, it is assumed that every node has the same maximum transmission range, and can adjust its transmission range depending on the packet type to be transmitted. If a node S transmits a packet type P, then the transmission range required for the transmission is denoted as TRS,P. If S uses its maximum transmission power, it is denoted as TRS,MAX or simply TRMAX.

Fig. 1 shows transmission ranges of S and V. A set of nodes in TRV is denoted as N(V). Any node P ∈ N(V) is called a one-hop neighbor (or simply a neighbor) of V, and it is said that there is a link connecting P and V. Likewise, if the minimum number of links connecting nodes P and Q is n, they are n-hop neighbors each other. In the following, a neighbor implies a one-hop neighbor, unless otherwise specified. In Fig. 1, nodes B, C, D are the neighbors of S, and D is also a neighbor of V. Note that transmission and one-hop broadcast are synonymous in wireless networks, since when the source node transmits a packet, the packet actually reaches all nodes in the source's transmission range (the same effect as one-hop broadcast). A communication algorithm is called reliable, if it completes its task. For example, if one-hop broadcast algorithm ensures that all nodes within the source node's transmission range receive the broadcast message, then the algorithm is reliable.

The broadcast threatening area of node V is an area such that the transmission from a node, say W, in that area causes TRV ∩ TRW ̸= {}. Since it is assumed in this paper

![Fig. 1. An example of the Hidden Terminal Problem. If both V and S are transmitting at the same time, there will be a collision at node D. Note that V cannot detect S's transmission since V is out of S's transmission range (S is hidden from V). The ring-shape dotted area centered at V with the radius 2 × TRV is the broadcast threatening area of V.](image)
that every node has the same maximum transmission range, $TR_V = TR_M = TR_{MAX}$ for every node. Therefore, the radius of the broadcast threatening area of V would be $2 \times TR_V$. For example, in Fig. 1, the ring-shape dotted area centered at V with $2 \times TR_V$ as the radius is the broadcast threatening area of V, because if any node, for example S, in the region is transmitting, it may cause a collision at a node within $TR_V \cap TR_S$.

In this paper, different channels are assigned to the different packet types. A channel that is used to transmit packet type P is denoted as $CH_P$.

### 2.2. Related works

Suppose node S in Fig. 1 is currently transmitting a packet. Also suppose node V wants to transmit a packet as well. If V knows S is transmitting, V will not start its own transmission, since it does, it causes collision at D. However, since node V is not within the transmission range of S, V cannot recognize that S is transmitting. Hence, V initiates its transmission, which causes collision at node D. This problem is known as the Hidden Terminal Problem [1].

To avoid the Hidden Terminal Problem, RTS–CTS protocol was proposed [16]. In RTS–CTS protocol, a transmitting node, say S in Fig. 2, has a data packet to send to D. Then, S sends a control packet called Ready-To-Send (RTS) to D prior to data packet transmission. The RTS contains the source ID (S in this example), the destination ID (D in this example), and the size of the data packet to be sent to node D. On receiving the RTS, D sends a control packet called Clear-To-Send (CTS) if it is ready to receive a data packet from S. The CTS also contains the source ID (S), the destination ID (D), and the size of the data packet from S to D. On receiving the CTS, S starts sending the data packet. An interesting feature of RTS–CTS protocol is that the CTS also reaches the nodes in D’s transmission range, for example node V. Therefore, from the information in the CTS, V calculates the duration of the transmission time of data packet from S to D, and does not transmit any packet during the time to avoid collisions.

Unfortunately, RTS–CTS is not a perfect solution for the Hidden Terminal Problem. The following two scenarios show the cases when RTS–CTS may not prevent collisions.

**Case (1)** Suppose there is a node B as shown in Fig. 2 whose transmission range contains V. If B is transmitting while D is transmitting CTS, the two packets are garbled at V. As a result, V cannot correctly decode the CTS from D, and may transmit a packet which causes collision at D with the data packet from S.

**Case (2)** Similar to the Case 1 above, but with the nodes C and S. Node C cannot decode RTS when node S transmits RTS. As a result, C may transmit a packet while D sends CTS to S, which causes collision at S. Therefore, S may restart the RTS–CTS process all over again.

In general, Case 1 is more serious problem than Case 2, because the size of data packets is bigger than that of control packets. Note that if all nodes in D’s transmission range (S’s transmission range, respectively) received the CTS from D (RTS from S, respectively) correctly, Case 1 (Case 2, respectively) would not occur. That is, if D’s (S’s) one-hop broadcasting of CTS (RTS) is reliable, there would be no collision at node D (S). The higher the traffic is, the more often the two cases described above would occur.

Note that RTS–CTS protocol is for point-to-point (or unicast) communications, i.e., communication between two nodes. However, if the communication is (one-hop) broadcast, the Hidden Terminal Problem causes more serious problem than unicast. That is because (1) S should exchange RTS–CTS with all its neighbors, but nodes in reactive routing algorithms do not maintain the list of all neighbor nodes, and (2) even if S has the list, it may take considerable amount of time to perform RTS–CTS to all nodes in the list, and (3) even during the execution of RTS–CTS, there may be new nodes moving into S’s transmission range and becoming its neighbors. Therefore, RTS–CTS may not be feasible for one-hop broadcast.

It is observed in Fig. 2 that transmissions from S to C and from D to V can be done simultaneously. However, since D and S are within each other’s transmission range, they cannot explore this simultaneous transmission if they use RTS–CTS protocol. This is an example of the Exposed Terminal Problem [2].

Park and Palasdeokar [8] have developed a reliable one-hop broadcasting algorithm called ROB that utilizes only one channel for both control packets and data packets. ROB does not require any control packet transmission prior to data packet transmission. When a node S transmits a data packet, if a node in S’s transmission range detects a collision, the node transmits a NACK control packet indicating that it did not receive the packet. Therefore, if S does not receive any NACK or collision, the one-hop broadcast could be regarded as completed. Otherwise, S continues transmitting the same data packet until it does not receive any NACK or collision. The algorithm is simple but gives rise to new problems such as Broadcast Complete Problem that presents another difficulty in detecting the completion of one-hop broadcast caused by NACKs.

Lembke et al. [9] proposed two reliable one-hop algorithms: one is proactive and the other is reactive. The reactive algorithm utilizes two channels and control packet called whistle which is similar to BLP in the proposed algorithm. In their algorithm, (1) nodes cannot adjust their transmission ranges, (2) nodes do not explore as many
simultaneous communications as possible to enhance the network throughput, (3) the approach to dealing with node mobility is limited and not efficient, and (4) there is a more serious problem, Whistle Propagation Problem, in which once a whistle is generated and propagated, it generates more whistles that live in the network forever causing the channels filled with useless whistles.

Park and Anderson [10] have proposed a guaranteed one-hop broadcasting algorithm called GOB that resolves the Broadcast Complete Problem [8] and Whistle Propagation Problem [9]. GOB uses a control packet BIP to prevent possible collisions, and therefore, improve network throughput. However, nodes in their algorithm do not explore simultaneous executions of one-hop broadcast either.

Although motivated by and uses similar terminology and concepts of the algorithms in [8–10], the proposed algorithm presents a significant improvement over the algorithms by exploring as many parallel executions of the one-hop broadcasting as possible, which produces far better performance in the network throughput.

3. Proposed algorithm: Efficient Reliable one-hop Broadcasting (EROB)

This section presents a new one-hop broadcasting algorithm, called Efficient Reliable one-hop Broadcasting (EROB) that guarantees the completion of one-hop broadcasting, i.e., all nodes in the source’s transmission range will receive the broadcast message.

Assumptions made in this paper are:

1. All nodes have the same maximum transmission range. Thus, all links could be bidirectional.
2. All nodes are capable of adjusting their transmission power, and therefore, their transmission ranges.
3. Each node implements multiple channels and is capable of switching between them.
4. Nodes receive correct data, unless there is a collision.
5. Each node knows its location and its moving speed using positioning devices such as GPS.
6. Moving speeds of nodes are moderate, otherwise, no algorithm would work.
7. Each node is capable of implementing CSMA/CA [31].

EROB uses the basic notation, definition and implementation similar to that of GOB [10], as described below. Two types of packets are used in EROB: control packets and data packets. Data packets contain the data to be one-hop broadcast, and control packets are used to enhance the efficiency of the data packet transmission. Although control packets may not be essential, the network throughput is usually higher with them due to their control over packet collisions. For example, using RTS and CTS control packets may produce higher network throughput [16].

EROB uses only a single type of control packet called Broadcast-In-Progress (BIP in short) to prevent collisions for achieving reliable one-hop broadcast. A BIP is produced and used in two cases:

1. Prior to one-hop broadcast of a data packet, the node transmits a BIP to secure not only the broadcast area but also broadcast threatening area as well to prevent possible collisions.
2. On receiving a BIP, a node that is currently involved in any other communication generates and transmits a BIP to warn other nodes in its broadcast threatening area not to initiate data packet transmission.

A BIP consists of unique bit pattern that distinguishes it from other types of packets. In addition to the actual data, a data packet contains other information such as the source’s ID, packet ID, and the size of the data packet.

For better understanding, EROB is presented in two sections: EROB with static nodes will be presented in Section 3.1, and node mobility will be added in Section 3.2.

3.1. EROB with static nodes

To reduce the probability of collisions, and to simplify the collision prevention process, EROB uses three different channels. CHBIP and CHDATA are dedicated to control packet (i.e., BIPs) and data packet transmissions, respectively. The third channel CHCOL is also for BIPs only but is used to prevent the BIP Propagation Problem that will be explained shortly. Since different types of packets use different channels, collisions can occur only between the same type of packets in the same channel, not between different types.

BIPs in EROB prevent data packet collisions as follows. Refer to Fig. 3. Recall that BIPs are transmitted only along CHBIP and data packets are transmitted only along CHDATA. Suppose node S in Fig. 3 has a data packet to be one-hop broadcast. Then, prior to data transmission, S prepares and transmits a BIP with TRMAX (=TRBIP) to inform the nodes in its transmission range with its intention of immediate data packet transmission. Then, on receiving the BIP, if a node is currently not involved in any communication, then the node remains silent. Otherwise, i.e., if the node is currently involved in any other communication, it transmits a BIP to warn S not to transmit data packet, since S’s transmission may cause collision at the node. On receiving either a BIP or a collision of BIPs, S refrains from transmitting data packet. Following two examples illustrate this case.

Case 1: Node in S’s transmission range is currently involved in data packet transmission. Suppose there is a node, for example D in Fig. 3a, in S’s transmission range that is involved in data packet transmission. Then, D prepares and transmits a BIP to warn S not to transmit data packet, since if S does, it would cause data packet collision at D. If there are two or more nodes that are involved in data transmission in S’s transmission range, for example C and D in Fig. 3b, they all transmit BIPs to warn S. In this case, although S would hear a garbled message (i.e., garbled BIPs) that is not possible to decode correctly, S interprets the situation correctly and not to transmit data packet.

Case 2: Node in S’s transmission range is currently involved in control packet transmission. Suppose there is more than one node, for example, S and V in Fig. 3 that
has data for one-hop broadcast. Then, as described above, they transmit BIPs prior to data transmission to prevent collisions. If there is only one node, e.g., D in Fig. 3a, in TRS,BIP ∩ TRV,BIP, then D hears a collision of BIPs along CHBIP. In this case, D realizes that the garbled signal received along CHBIP is due to the simultaneous reception of BIPs, and prepares and transmits a BIP to warn both S and V for possible collision. On receiving the BIP, both S and V do not transmit data packets. The same process is applied to the case when there is more than one node, e.g., C and D in Fig. 3b, in TRS,BIP ∩ TRV,BIP, except that this time both S and V hear garbled signal, instead of a BIP. In summary, whether S hears a BIP or collision, it would not transmit data packet.

Although BIPs are very useful, they also may cause a serious problem, called BIP Propagation Problem, similar to Whistle Propagation Problem [9]. To demonstrate the problem, refer to Case 2 above. When C and D in Fig. 3b receive garbled BIPs, one from S and one from V, they transmit a BIP to warn both S and V for possible collision. Observe that the BIPs from both C and D cause BIP collisions at nodes E and F. On receiving the garbled BIPs, nodes E and F transmit BIPs that, in turn, cause a collision at node G as well as C and D. This phenomenon goes on and on, and eventually the network is filled with BIPs, preventing nodes from doing useful work. The approach EROB has taken to resolve the BIP Propagation Problem is to use additional channel CHCOL as described below.

When a node detects a BIP or a collision of BIPs in CHBIP, it transmits a BIP only along CHCOL indicating that the BIP is the response to the first BIP collision. For example, nodes C and D in Fig. 3b transmit a BIP only along CHCOL when they detect a collision in CHBIP caused by two BIPs from S and V. When E and F receive the garbled BIP along CHCOL, they know it is not the first BIP collision because it is along CHCOL, and do not transmit any BIP. Therefore, the BIP is dropped at C and D, and is not propagated to G as well as not back to C and D. Note that regardless of the reception along CHCOL, if a node detects a collision in CHBIP, the node transmits a BIP along CHCOL.

Good communication algorithms should provide advantages such as (1) power conservation in packet transmission, (2) low probability of packet collision, and (3) better network throughput by allowing as many simultaneous transmissions as possible. In order to achieve these, a node in EROB uses different transmission power levels depending on its current status (that is, either sending or receiving) and the types of the packet it transmits. Data transmitting nodes use TRMAX (maximum power level) for BIP transmission and TRmax for data packet transmission. On the other hand, data receiving nodes transmit BIPs, if necessary, with TRmin. Following example shows how EROB accomplishes better network throughput by controlling the transmission range.

Refer to Fig. 3b. If both S and V transmit data packets with TRMAX, there will be a collision at C and D. One possible solution is to let node D, on receiving the BIP, prepare and transmit its own BIP with TRMAX so that both S and V hear it and refrain from transmitting data packets. This solution is simple, but not productive since no node can transmit any data. The approach EROB takes to improve network throughput is to reduce the transmission range for data packets to TRDATA = TRmin = TRmax. With this transmission range, there will be no overlapped area, i.e., TRS,DATA ∩ TRV,DATA = ∅, so that both S and V may be able to transmit their data packets at the same time without causing any collision. Our simulation results show that the network throughput has been improved considerably with this various transmission ranges.

Incorporating the schemes described above, EROB at each node executes the following phases.

Phase (1) Prior to one-hop data packet broadcast, S listens to the channels, CHBIP, CHDATA, and CHCOL for any ongoing transmission. Then, the following two cases can occur.

Case (1) S detects on-going transmission in any of the three channels. In this case, if S starts transmitting, it may cause collision. Therefore, S waits a random amount of time, and then starts Phase 1 again. The average random waiting time can be $\frac{1}{2}$ (BIP transmission time + average data packet transmission time). Note that since the size of the data packet may vary, and since S does not know the size of the on-going data packet, the default value should be a reasonable value.

Case (2) S does not hear any on-going packet transmission in any channel. In this case, S enters Phase 2.

Phase (2) S transmits a BIP packet (designated by (1) in Fig. 3) along CHBIP with the maximum power TRMAX to inform its neighbor nodes with its intention of immediate

![Fig. 3. Examples of how BIPs are used to prevent collisions. (a) only one node in TRS,BIP ∩ TRV,BIP, and (b) more than one node in TRS,BIP ∩ TRV,BIP.](image-url)
transmission of data packet. During the BIP transmission, if S detects any on-going transmission except its own transmission in any of three channels, then S immediately stops transmitting the BIP and enters Phase 1. Otherwise, after transmitting the BIP, S enters Phase 4. All nodes in TRS,MAX enter Phase 3.

**Phase (3)** A node in TRS,BIP either receives a BIP or detects a collision along CHBIP, depending on the number of BIP transmitting nodes within its TRMAX. For example, suppose both S and V in Fig. 3 transmit BIPs simultaneously. Then, node F hears the BIP from S, but D detects a collision since it receives two BIPs from S and V. Depending on the current status, a node, say N in TRS,BIP, N ≠ S, falls into one of the following three cases.

Case (1) N is currently transmitting a data packet. Note that N cannot be within \( \frac{1}{2}\text{TR}_{\text{MAX}} \) (≈ TRS,DATA), since if it is, S must have heard the data packet transmission from N in Phase 1 and could not enter Phase 2. Therefore, in this case N must be in the broadcast threatening area of S as shown in Fig. 4a. If S starts transmitting data packet, there will be a data packet collision at nodes, for example C, in TRS,DATA ∩ TRS,DATA. Therefore, to avoid any data packet collision, N immediately transmits BIP along CHCOL with power level of TRMAX to warn S not to transmit. If N is receiving a signal along CHCOL only, this implies that N is out of TRS,MAX. Because the signal is to prevent BIP propagation, N ignores the signal and continues transmitting the data packet.

Case (2) N is currently receiving a data packet. In this case, the location of N could be anywhere within TRS,BIP (=TRMAX). To avoid a collision, node N transmits a BIP along CHCOL with power level of \( \frac{1}{2}\text{TR}_{\text{MAX}} \). Then, depending on the location of N, the following two subcases can occur.

Subcase (1) N is within \( \frac{1}{2}\text{TR}_{\text{MAX}} \) (≈ TRS,DATA). Assume that N is currently receiving data packet from node, say V, as shown in Fig. 4b. (Note that V is within the broadcast threatening area of S, and cannot be within \( \frac{1}{2}\text{TR}_{\text{MAX}} \), since if it is, S must have heard V transmitting data during Phase 1 and could not enter the Phase (2).) If S transmits data packet, there would be a collision at node N. Therefore, BIP transmission from N with transmission power level of \( \frac{1}{2}\text{TR}_{\text{MAX}} \) will prevent S from transmitting data. As a result, there will be no collision.

Subcase (2) N is within S’s broadcast threatening area (i.e., between \( \frac{1}{2}\text{TR}_{\text{MAX}} \) and TRS,MAX) as shown in Fig. 4c. In this case, transmission of BIP from N with power level of \( \frac{1}{2}\text{TR}_{\text{MAX}} \) would not reach S, which allows simultaneous transmissions from both S and V without collision. This may enhance the network throughput.

Case (3) N is not involved in any data packet transmission. In this case, N can be one of the following three subcases.

Subcase (1) N is receiving a single BIP in CHBIP. This implies that there is only one node (S in this case) that is currently transmitting in N’s transmission range. Since there is no possibility of collision, N keeps silent and does not do anything.

Subcase (2) N is detecting a collision of BIPs in CHBIP. This implies there is more than one node trying to transmit in N’s transmission range, which may cause collision at N. To prevent the collision, N transmits a BIP along CHCOL with the power level of \( \text{TR}_{\text{MAX}} \) to warn the nodes not to transmit.

Subcase (3) N is receiving a BIP or collision along CHCOL only. Any signal along CHCOL is to prevent BIP Propagation Problem. Therefore, N ignores the signal and keeps doing what it is currently doing.

**Phase (4)** After transmitting a BIP, S waits for response, if any. Then two cases can occur.

Case (1) S detects a BIP or a collision of BIPs in CHCOL. In this case, S stops and enters Phase 1.

Case (2) S does not detect any signal in any channel. In this case, S can go ahead and transmits its data packet with TRMAX. During the data transmission, if S detects a BIP or a collision in CHBIP, it will transmit a BIP along CHCOL with power level \( \text{TR}_{\text{MAX}} \) immediately to suppress the attempt.
of transmission from other nodes that may cause collision (in this case, S is N in Case 1 of Phase (3). Also, during the data transmission, if S detects data packet collision, it immediately stops transmitting data packet, and enters Phase 1 (Note that this case may not happen, since EROB in this section assumes no node mobility).

All other nodes that detect a collision of BIPs in CHCOL remain silent since the collision is the result of BIP propagation.

Fig. 5 presents the summary of EROB. Section 3.2 presents the improvement of EROB in the presence of reasonable node mobility.

### 3.2. EROB with node mobility

EROB described in Section 3.1 accomplishes one-hop broadcasting provided that the nodes in the network are stationary. Therefore, if nodes become mobile, there might be more possibility of data packet collisions than the cases described in Section 3.1. For example, suppose nodes S and V in Fig. 6a have data packets for one-hop broadcasting. Note that if both S and V implement EROB described in Section 3.1, then there will be no data packet collisions when they start transmitting data packets, since there is no overlapped area of TRS\_DATA \cap TRV\_DATA. However, if S and V are moving towards each other while transmitting data packets, an overlapped area will be formed eventually and may grow bigger, as shown in Fig. 6b. This will cause data packet collision at the nodes, for example C in the overlapped area. Note that if there is a “enough distance” between TRS\_DATA and TRV\_DATA at the beginning of the data transmission, there would be no collision. This observation had led us to develop the following data collision avoidance scheme that guarantees the completion of one-hop broadcasting even in the presence of node mobility.

Although the discussion in this section focuses on data transmission, it could also be applied to transmission control as well. Suppose nodes S and V in Fig. 6a just started transmitting data packets while moving towards each other with the speed of SPEED\_S and SPEED\_V, respectively. Also, let TIME\_S and TIME\_V be the data packet transmission time of S and V, respectively. Note that this information is hidden from each other, because they even do not know the existence of each other. Our approach to avoid the possible collision is to give enough distance in advance between TRS\_DATA and TRV\_DATA so that they would not overlap during the data transmission. That is, if the distance between S and V is greater than SPEED\_S \times TIME\_S + SPEED\_V \times TIME\_V when they start transmitting data packets, there would be no collision. Since we may not be able to control the mobility of nodes, the only way to secure the distance to prevent the collision would be adjusting the transmission range. Therefore, if S reduces its transmission range TRS\_DATA by D\_S = SPEED\_S \times TIME\_S, data packet collision could be avoided from S’s side. However, since both S and V are moving towards each other in the worst case, V should also reduce its transmission range TRV\_DATA by D\_V = SPEED\_V \times TIME\_V. Then, there will be no data packet collision, unless either SPEED\_S or SPEED\_V increases.

---

**Algorithm EROB**

<table>
<thead>
<tr>
<th>At the broadcasting node S</th>
<th>At the receiving node D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. S listens its surrounding for any on-going transmission. If there is, S waits a random amount of time and starts step 1. Otherwise, goes to step 2.</td>
<td>3. D received either a BIP or a collision along CH_BIP. Regardless of what it received, D performs one of the followings depending on its current status.</td>
</tr>
<tr>
<td>2. S transmits a BIP via CH_BIP and waiting for any response. If S detects a BIP or BIP collision during the transmission, it stops transmission, and goes to step 1.</td>
<td>Case 1: D is transmitting a data packet. D transmits a BIP along CH_COL with TR_MAX.</td>
</tr>
<tr>
<td>4. If S hears any in CH_COL, it waits a random amount of time, and goes to step 1. Otherwise, S starts transmitting data packet. If S hears a BIP or collision of BIPs along CH_BIP during data transmission, it transmits BIP along CH_COL with TR_MAX.</td>
<td>Case 2: D is receiving a data packet. D transmits a BIP along CH_COL with TR_MAX \div 2.</td>
</tr>
<tr>
<td></td>
<td>Case 3. D is not involved with data packet transmission. D keeps silent.</td>
</tr>
</tbody>
</table>

---

Fig. 5. summary of EROB.
In summary, when a node $N$ starts transmitting data packet, it adjusts its transmission range to $TR_{N, DATA} = \frac{TR_{\text{MAX}}}{2} - \text{SPEED}_N \times \text{TIME}_N$. This gives the additional distance between transmission ranges to prevent the collisions due to mobility.

4. Simulation

In our simulation, environment similar to GOB [10] is used. Three algorithms have been tested and compared in the simulation. The first algorithm is naïve one without any prevention scheme on collision, and as soon as a node has a data packet to send, it sends it right away. The second algorithm uses CSMA/CA protocol in which a node starts transmitting data packets only if there is no on-going transmission. Otherwise, it waits a random amount of time and tries again. Third algorithm is EROB.

The size of the network in our simulation is $2000 \times 2500$, and transmission range is 300. At the beginning of simulation, each node is assigned random starting and destination positions with randomly chosen speed of between 0 and 100. Once a node reaches its destination, the destination becomes new starting point and the new destination is assigned with new speed. Nodes move straight between starting and destination point. In our simulation, it is assumed that every node always tries to one-hop broadcast to generate hostile environment. Data packet transmission duration is 10 and BIP is 2, which seems reasonable since BIP size is much smaller than data packet. Each simulation lasts 200 unit times.

First simulation has been performed to discover the relationship between the number of nodes and the number of collisions generated by three algorithms. The result is shown in Fig. 7. The number of collisions by naïve algorithm increases quite rapidly, and the number of collisions reaches about 40,000 when the number of nodes in the network is 200. On the other hand, CSMA/CA and EROB perform much better than naïve algorithm. To be more specific, EROB outperforms CSMA/CA by factor of about 7 when the number of nodes is close to 200.

Second simulation is to find out the relationship between the number of nodes in the network and the number of successful transmissions of the three algorithms, and the result is shown in Fig. 8. The success ratio of EROB becomes much better than the other two as the number of nodes in the network increases.

Fig. 9 is the result of the third simulation that shows the relationship between the packet size and the number of successful transmissions of the three algorithms. Speed and node size are fixed to 100 and 100, respectively. Again, EROB outperforms Naïve and CSMA/CA algorithms by the factor of at least 5. Our simulation results clearly show the superiority of EROB in every aspect.

5. Conclusion

One-hop broadcast is a fundamental communication primitive in which the source node delivers its packet to all nodes within its transmission range. One-hop broadcast is very useful in almost all networks, especially in wireless networks where every transmission is a one-hop broadcast by nature. Despite the importance of it, it is hard to accomplish in wireless networks due to the collisions caused by Hidden Terminal Problem.
This paper presents an algorithm, called Efficient Reliable One-Hop Broadcast (EROB in short), that guarantees the completion of one-hop broadcast in wireless mobile ad hoc networks. In addition to data packets, EROB uses a single type of control packets, Broadcast-In-Progress (BIP), to prevent collisions. EROB also implements three different channels, one for data packets and the other two for BIPs to prevent collisions further. Another unique feature of EROB is to allow each node to adjust its transmission range so that as many simultaneous one-hop broadcasts be explored as possible to enhance the network throughput. Other advantages obtained from the adjustment of transmission ranges include (1) power saving due to smaller transmission range, (2) less number of collisions, and (3) longer network lifespan.

Simulation results show the significant improvement of the proposed algorithm over the algorithms that either take no precaution on collisions or implement a simple scheme to avoid collisions.

References
