Dual Covered Broadcast with Negative Acknowledgements DCN : A Broadcast Algorithm for Mobile Ad Hoc Networks

Thriveni J, Ashwini B, Latha A, Sandyashree K R, K R Venugopal, L M Patnaik *†‡§

Abstract—**Broadcasting in Ad hoc networks, widely utilized as a building block for many network layer protocols is prone to broadcast storm problem. This problem necessitates to carefully designate some nodes in the one-hop neighborhood of the sender as forwarding nodes and reduce broadcast redundancy. In this paper, we propose an algorithm called Dual Covered broadcast with Negative acknowledgements (DCN) which focuses on achieving high delivery ratio in an environment that has high transmission error rate. We make use of Negative ACKnowledgements (NACK) to reduce broadcast collision and achieve enhanced reliability. Overhead caused due to Acknowledgements are avoided by allowing the sender to overhear the transmissions from the receiver. Broadcast congestion is alleviated by eliminating the unnecessary duplication of packets, otherwise resulting in Broadcast storm problem. Simulation results show that DCN algorithm improves the reliability of broadcast operation compared to Double Covered Broadcast (DCB) and use lower number of forwarding nodes in the network.**

Keywords:* *Broadcast, forwarding node, reliability, Mobile Adhoc NETworks (MANETs), NACK (Negative ACKnowledgement)

1 Introduction

A Mobile Ad hoc network consists of a group of mobile nodes forming a temporary network on wireless links without the aid of any centralized administration. Some of its characteristics are: dynamic topology, bandwidth constraint, energy constraint, limited physical security. Ad hoc networks are used in rescue operations, disaster recovery, hospitals, conferencing, communication, military etc.. In order to facilitate communication within the network, a routing protocol is used to discover routes between nodes. This is one of the functionality provided by broadcast in MANETs. Broadcasting a message, originating from a source node, to all the nodes in the network need the support of intermediate nodes. Selection of intermediate nodes that relay the messages is of key concern in broadcast operation.

Broadcasting nature of radio transmission may cause several problems such as *exposed terminal problem* or *hidden terminal problem*. Exposed terminal problem causes an outgoing transmission to collide with an incoming transmission and hidden terminal problem causes two incoming transmissions to collide with each other. Broadcast protocols are classified into following categories: flooding, probability based broadcasting, area based broadcasting and neighbor knowledge based broadcasting. Probability based broadcasting is similar to flooding except that nodes only rebroadcast with a predetermined probability. This scheme works for dense networks. It does not provide reliable broadcast for sparse network. In area based methods, nodes are assumed to have common transmission distances. A node rebroadcasts only if it can reach sufficient additional coverage area.

When a mobile host broadcasts a message and if most of its neighbors decide to rebroadcast the message, then these transmissions may severely cause network congestion resulting in too many redundant packets in the network. This is referred to as *Broadcast storm problem*. The congestion caused due to this problem is reduced by designating only a subset of nodes as forwarding nodes to forward the message and ensure that the non-forwarding nodes adjacent to the forwarding nodes receive the message to achieve broadcast coverage. Selection of forwarding nodes in either dense or sparse network should be such that the density of the network is reduced. Forwarding nodes that satisfy the above mentioned criteria form a *connected dominating set*.

Motivation: A major challenge in dynamic MANETs

*Thriveni J is an Assistant Professor and Chairperson, Department of Computer Science and Engineering, University Visvesvaraya College of Engineering, Bangalore University, Bangalore and a Ph. D student at Dr. M.G.R. University, Chennai e-mail: thrivenijgowda@yahoo.co.in.

†Ashwini B, Latha A and Sandyashree are Undergarduate students in the Dept. of CSE, UVCE, Bangalore.

‡Dr. K R Venugopal is a Professor in the Dept. of CSE and is the Principalof UVCE, Bangalore University, Bangalore.

§Dr. L M Patnaik is the Vice Chancellor of Defence Institute of Advanced Technology, Pune, India.

is to ensure high reliability for broadcast operations inspite of high transmission error rate. Usually ACKs are used to ensure broadcast delivery. However, the requirement for all receivers to send ACKs in response to reception of a packet may become another bottleneck of channel congestion and packet collision. This situation is referred to as *ACK implosion problem*. All these issues need to be considered in achieving broadcast reliability in MANETs. Though flooding is simple, it consumes much network resources as it introduces large number of duplicate messages. This leads to serious redundancy, contention and collision in MANETs. Flooding in MANETs has poor scalability. Due to above mentioned problems in different broadcast protocols, we implement a broadcast routing protocol called Dual Covered Broadcast to achieve reliability which is based on Neighbor knowledge based broadcasting.

Contributions: In this paper, we propose an algorithm called Dual Covered Broadcast algorithm which aims at reducing the number of forwarding nodes thereby reducing packet congestion and collision and still achieve high broadcast delivery ratio. The selected forwarding nodes should be such that they cover sender's two-hop neighbors and sender's one-hop non-forwarding neighbors atleast twice. The retransmissions of forwarding nodes are received by sender as acknowledgement of their reception of the packet, solving ACK implosion problem.

Organization: The remaining part of the paper is organized as follows: Related work is reviewed in Section 2. Section 3 describes the Network Model and Background. Problem Definition and Algorithm is given in Section 4. Section 5 gives the Performance Evaluation. Conclusions and Future Work are contained in Section 6.

2 Related Work

Min Sheng et al., [1] have proposed Relative Degree Adaptive Broadcast algorithm for efficiently reducing the broadcast overhead in the network. Based on the current states of the network and degree of the nodes, algorithm computes the relative degree of the nodes, decides and determines the nodes that need to re-transmit and the nodes that only need to receive. Simulation results show that RDAB strategy outperforms the ordinary flooding method and multi-point relaying protocol for Ad Hoc Networks. Mansoor et al., [2] have addressed a single source reliable broadcasting algorithm for linear grid based networks where a message is guaranteed to be delivered to all the nodes of the network. The protocol takes into consideration node mobility multiple nodes located at the same point. When there is only a single broadcasting source, the protocol presented is energy-efficient, has low latency and is collision free.

Jie Wu et al., [3] have developed an approach that chooses a subset of nodes, called forward node set, to relay the broadcast packets. Each clusterhead computes its forward node set that connects its adjacent clusterheads. Simulation results show that its performance improvements against other broadcast algorithms. Shen Jun et al., [4] have proposed a new broadcast protocol, Receiver-deciding Location-Aided Broadcast Protocol. Three other well-selected broadcasting protocols, i.e., simple flooding, Ad hoc Broadcast Protocol, are also evaluated via simulation in comparison with this algorithm. The algorithm provides higher delivery ratio in network congestion conditions. It has shorter end-to-end- delay and consumes less network resources.

Fei Dai et al., [5] have addressed a general framework for broadcasting in Ad Hoc networks through self pruning. Each node, upon receiving a broadcast packet, determines whether to forward the packet based on two neighborhood coverage conditions. The forward node set can be constructed and maintained through either a proactive process or a reactive process. The self pruning scheme in general, is more efficient in reducing the forward node set than several existing schemes. Song et al., [6] have developed a scheme to improve the reliability and efficiency of the broadcast protocol in MANET. This scheme improves delivery rate with minimum overhead without degrading efficiency. The proposed scheme *Relayer Broadcast Sequence Number* allows sender to identify the passing packets, minimizes packet loss and improves reliability by confirming the broadcast packet reception based on Negative Acknowledgements.

Wei Lou et al., [7] have analyzed some deficiencies of the dominant pruning algorithm and have proposed two algorithms, Total Dominant Pruning and Partial Dominant pruning. Both algorithms utilizes two hop neighborhood information more effectively to reduce redundant transmissions. The result of applying these two algorithms show performance improvements compared with the original dominant pruning. Ivan et al., [8] have proposed an algorithm to significantly reduce or eliminate the communication overhead of a broadcasting task by applying the concept of localized dominating set. Re-transmissions by only internal nodes in a dominating set is sufficient for reliable broadcasting. The authors proposed to eliminate neighbors that already received the message and rebroadcast only if the list of neighbors that might need the message is nonempty.

Qayyum et al., [9] have proposed selected Multipoint Relays i.e., forward nodes to propagate link state messages in their Optimized Link State Routing protocol. The Multipoint Relays are selected from 1-hop neighbors to cover 2-hop neighbors. Forwarded nodes are not considered for a node to select its successors and, therefore, the entire set of neighbors must be covered. The basic categories of reliable communication schemes are *sender initiated* and *receiver initiated* approaches [10]. In the sender initiated approach [11], [12], the receiver returns a positive Acknowledgement to the sender for each message it receives. The drawback of this scheme is that the sender may become the bottleneck of transmission when simultaneous Acknowledgements return. The amount of records that the sender must maintain may grow large. In the receiver initiated approach [13], [14], the receiver is responsible for reliable delivery. Each receiver maintains receiving records and requests repairs via a negative acknowledgement when errors occur.

Shue et al., [15] have developed several reliable broadcast schemes that aim to suppress MAC layer's collision and provide reliable transmission. In the network layer, most reliable broadcast protocols come from the routing protocol proposed by [16]: the source starts a broadcast operation by sending a message to all its neighbors and waiting for the ACKs from its neighbors. When it receives all these Acknowledgements, it sends the message asking the neighbors to propogate the message one more hop to their own neighbors. The neighbors of the source forward the message to their neighbors and send the Acknowledgements back to the source when they receive all Acknowledgements from all their own neighbors and so forth. This scheme incurs too much communication overhead and needs stable linkages for ad hoc networks.

Garcia et al., [17] and [18] have proposed a flooding based reliable broadcast protocol that allows the nodes that received the broadcast packet to forward the packet without further notice from the sender. The drawback of this protocol is that the flooding may easily introduce the broadcast storm problem. The Acknowledgement implosion problem may worsen the broadcast storm problem. Pleisch et al., [19] have addressed an approach that relies on proactive compensation packets to overcome low level residual packet losses. Pagani et al., [20] have proposed to setup a forwarding tree, which is routed from the clusterhead of source to each clusterhead, based on a virtual cluster architecture for a reliable broadcast in ad hoc networks.

Wei Lou et al., [21] have proposed a simple broadcast algorithm, called Double-Covered Broadcast which takes advantages of broadcast redundancy to improve the delivery ratio in an environment that has rather high transmission error rate. Among the one-hop neighbors of the sender, only selected forwarding nodes re-transmit the broadcast message. Simulation results show that the algorithm provides high delivery ratio, low forwarding ratio, low overhead and low end-to-end delay for a broadcast operation under a high transmission error rate environment.

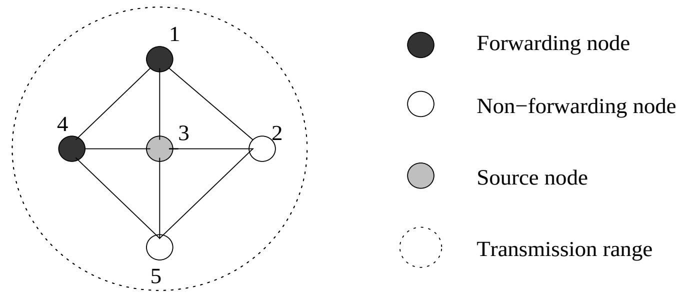

Figure 1: Illustration of a Wireless Mobile Ad Hoc Network

3 Model and Background

3.1 Network Model

Let ad hoc network be a unit disk graph $G = (V, E)$ as shown in Figure 1, where the node set V represents a set of wireless mobile nodes and edge set E represents a set of bi-directional links between the neighboring nodes. Two nodes are considered neighbors if and only if their geographic distance is less than the transmission range r. For a node v, k-hop subgraph is denoted by $G_k(v)$ for a small k such as $k = 2$. All nodes in the network has 2-hop neighbor information. $G_k(v)$ induced from k-hop information of v, is $(N_k(v), E_k(v))$. $N_k(v)$ denotes the k-hop neighbor set of node v which includes all nodes within k-hops from v. For a specific node, the upstream node that has sent a broadcast packet to this node is viewed as a *forwarding node*. A *forward node* is a downstream node designated by this node that forwards the broadcast packet. A *non-forward node* is a downstream node that is designated not to forward the packet. Metrics used for performance evaluation of DCN protocol are as follows:

(i) *Broadcast delivery ratio* : It is the ratio of the number of the nodes that received packets to the number of nodes in the network for one broadcast operation.

(ii) *Broadcast forwarding ratio* : It is the fraction of the total number of nodes in the network that atleast retransmit broadcast packets once for one broadcast operation.

(iii) *Broadcast overhead* : It defines the ratio of the total transmissions, including the broadcast packets and extra control packets such as HELLO and ACK messages, to the broadcast packet per node. It is measured by bytes per broadcast byte per node.

(iv) *Broadcast end-to-end delay* : It measures the period from the time the source broadcasts the packet to the time the last node receives the packet or no more nodes resend the packet for one broadcast operation.

Table 1: **Notations used in the Algorithm**

symbols	*definition*
F(v)	Forward node set of node v
U(v)	Uncovered 2-hop neighbor set of v
C_v(p)	Counter to track number of times a packet p sent by node v
R_timer	Timer for resending the packet
T_wait	Bound to overhear retransmissions of senders forward nodes
P(s,v,F(v),N(v))	Broadcast packet P from a source s, attaching F(v), N(v)forwarded by v
RT_{max}	Number of retries for data packets
H_timer	A timer for sending hello messages
T_hello	Bound on the timer for a node to send hello messages
Nb_timer	Timer for refreshing neighbor info.
T_nb	Bound on the timer to refresh neighbor information
Nk_timer	Timer for sending NACK messages
T_nk	Bound to send NACK messages
MAX_NACK	Number of retries for NACK messages

3.2 Background

In a broadcast operation, a source node disseminates the packet to all the nodes in the network. Due to interference of transmission of neighbors or mobility of nodes, some nodes may not receive the packet. We ensure reliability by taking advantage of broadcast redundancy. Unlike flooding, Double Covered Broadcast algorithm designates only few nodes as forwarding nodes that can forward the packet and achieve reliability. Table 1 shows the notations used in the algorithm.

In DCB algorithm, a node can play two different roles: *forwarding node* and *non-forwading node*. Functions of a forwarding node are:

(i) It records the packet sent by the upstream node.

Table 2: Algorithm Forward Node Designating Process

```
FNDP (node v)
begin
    Let X(v) = N(v) - {v}, U(v) = N_2(v) - {v}
    and F= ø
    if (node v ≠ source node) then
        X(v) = X(v) - F(u) - {u} - {source}
        U(v) = U(v) - N(u) - N( (F(u)-v) ) - N(source)
    end if
    while ( u ≠ ø )
        Find node w in X with the maximum effective
        degree Deg_e(w) = | N(w) ∩ U |
        F = F ∪ {w}, U = U - N(w) and X = X - {w}
    end while
end
```

(ii) It selects some nodes in its 1-hop neighborhood as forwarding nodes to forward the packet that satisfy two selection criteria:

(a) selected nodes should cover maximum number of nodes in the 2-hop neighborhood of sender.

(b) non-forwarding nodes in the 1-hop neighborhood of sender should be covered atleast twice.

(iii) It maintains a list of forwarding nodes and rebroadcasts the packet as a new sender.

(iv) The retransmission of the forwarding nodes are received by new sender as acknowledgement.

(v) It waits for predetermined duration to overhear the retransmissions from its forwarding nodes.

(vi) If it fails to detect all its forwarding node's retransmissions during this duration, then it resends the packet until its forwarding nodes retransmit or maximum number of retries is reached.

Table 2 shows the Forward Node Designating Process (FNDP) algorithm used to select the forwarding nodes as described above.

Supposing u is the last forwarded node and v is a designated forwarding node of u, v selects its forwarding node set F from candidate neighbors in X(v) to cover its uncovered 2-hop neighbors in U(v) with a greedy algorithm FNDP. Node v updates U(v) by excluding N(u), N((F(u)-v)) and N(s) because N(u) are the neighbors of u which is covered by u. N((F(u)-v)) denotes the set of neighbors of forwarding nodes of u excluding v. This set of nodes is covered by forwarding nodes of u excluding v. N(s) denotes the neighbor set of source which is covered by source node. X(v) is updated

Table 3: Actions taken at source node

```
When a source s originates a packet:
begin
    FNDP(s)
    C_s(p) = 0
    R_timer = T_wait
    s broadcasts P(s,s,F(s),N(s)) among N(s).
end
```

by eliminating forwarding nodes of upstream sender u, node u and s. This is done to eliminate the possibility of choosing same forwarding node many times. Every time a forwarding node is selected using FNDP, neighbor nodes of the selected node are eliminated from U-list and selected node is eliminated from X-list.

A non-forwarding node does not forward the packet further. It only records the packet and does not acknowledge the packet reception. If the non-forwarding node does not receive any packet, then it sends NACK message requesting the missing packet. The algorithm is named so because of the selection criteria (b) mentioned above. The algorithm does not suffer from the disadvantage of the receiver-initiated approach that needs a much longer delay to detect a missed packet.

3.3 Reliability issues

When a sender transmits a packet to all its neighbors, a neighbor may fail to receive this packet because of a transmission collision with other neighbors, the high transmission error rate of the radio channel or the out-of-range movement of the node.

We treat the non-forwarding node and forwarding node differently. When a non-forwarding node v fails to receive the packet (Figure 2(a)), based on the FNDP, v has been atleast covered by two forwarding nodes u and f; even when v fails to receive the packet from u, it still has a second chance to receive the packet from f. If the non-forwarding node v fails to receive the packet even for the second time, then it uses NACK messages to receive the lost packet. Note that a non-forwarding node that failed to receive the packet does not cause other transmission error prorogation in the network. When a forwarding node f fails to receive the packet, it may cause the transmission error to propagate since forwarding nodes are the key nodes in the network that need to relay the broadcast packet. There are two main causes for the packet loss;

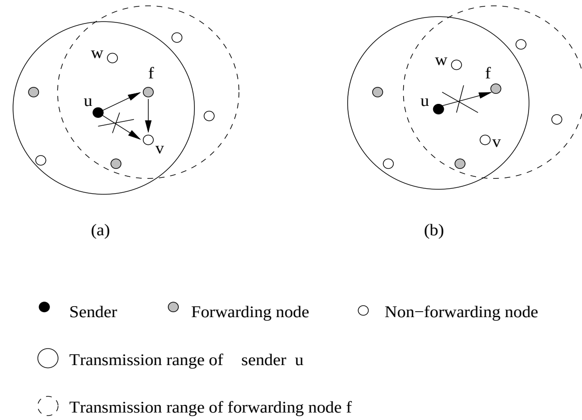

Figure 2: Illustration of transmission errors: (a) a transmission error occurs at a non-forwarding node v. (b) transmission error occurs at a forwarding node f that causes nodes in the transmission range of f to miss the packet.

Transmission collision and high transmission error rate; In Figure 2(b), if f does not receive the transmission from u because of the transmission collision or transmission error of the radio channel, the nodes in the transmission range of f may miss the packet. The simple strategy to send packets is adaptive to this case: u waits for a period of time T_wait when it sends a broadcast packet. If u fails to detect f's retransmission signal during T_wait, u sends the packet repeatedly until the maximum retry is reached.

Out-of-range movement of the node: A selected forwarding node may move out of sender's range and this results in a transmission failure. For this scenario, the node which has missed the packet of sequence number n, sends NACK request for that packet after receiving $(n+1)^{th}$ packet. This works for both forwarding and non-forwarding nodes.

4 Problem Definition and Algorithm

Given a wireless MANET with N number of nodes, a broadcast operation encounters problems like Acknowledgement implosion and Broadcast storm problems. These problems reduce broadcast delivery ratio because of network congestion and packet collision, which inturn affects broadcast reliability. ACKs, originally meant for ensuring reliability now becomes an obstacle for achieving reliability.

4.1 Objectives

(i) To solve ACK implosion problem and ensure broadcast reliability without using ACKs to confirm the reception of packet.

(ii) To solve Broadcast storm problem by carefully selecting few nodes to forward the broadcast packet.

(iii) To achieve reliability with low end-to-end delay and broadcast overhead.

4.2 Assumptions

(i) Network is assumed to be connected.

(ii) Omni-directional antennas are used.

(iii) All nodes are assumed to have symmetric links i.e., same transmission range.

(iv) Low mobility scenarios are considered.

4.3 Algorithm

When a node s starts a broadcast process, it uses FNDP algorithm to select its forwarding node set F(s) and piggybacks F(s), N(s) with the packet. It broadcasts the packet among its 1-hop neighbor set N(s). The *hello* and *neighbor* timers for the source node are scheduled to exchange hello messages and refresh neighbor information periodically. Forward timer is also scheduled to ensure that the node s overhears retransmissions from all its forwarding nodes. Table 3 shows the actions taken by a source node.

When a node v receives packet P from an upstream node u, it records P and extracts the neighbor information from the packet. Node v examines whether any NACK timer for the received packet has been scheduled. If so, it cancels the NACK timer. Node v also checks whether the previous packet has been received. If not, it makes sure that sending of NACK message does not cause flooding and then it schedules the NACK timer and sends NACK message. It waits for a predetermined duration to receive the missed packet as reply. Sending NACK request for a particular packet is limited for MAX_NACK times. If node v is designated as forwarding node by node u then it checks whether the received packet is a new packet or a duplicate packet.

If the packet is new, node v uses FNDP algorithm to select forwarding nodes F(v) to cover nodes in the set U(v). Node v broadcasts the packet among its 1-hop neighbors N(v), instructing the forwarding nodes to forward the packet further. If the packet is a duplicate packet, then the receiving node v locally broadcasts the packet with the forwarding node list set to null.

When a node v receives a packet P from an upstream node u, it checks whether node u belongs to forwarding node set of v i.e., u∈F(v). If so, node v updates F(v) by eliminating node u. Node v assumes that it has received the packet as an acknowledgement from node u. If F(v) becomes null within the duration T_wait, then the R_timer gets canceled. If node v is not a forwarding node of u, then it only records the packet. Irrespective of whether a node is a forwarding node or a non-forwarding node, NACK messages are used to ensure reliability. Table 4 shows DCN algorithm.

Forward timer, denoted by R_timer, is set at every forwarding node v to overhear retransmissions from its forwarding nodes F(v). This timer times out after an interval of T_wait. If node v overhears retransmissions from all its forwarding nodes within this time interval, it cancels the timer. Otherwise, it checks whether the node from which node v has not received the retransmission is still in its 1-hop list. If so, it increments the counter

Table 4: Algorithm Dual Covered Broadcast with NACK

```
DCN( )
begin
    When a node v receives P (s, u, F(u), N(u)):
    if ( v ∈ F(u) ) then
       if ( P is received for first time ) then
          FNDP(v)
          v broadcasts P (s, v, F(v), N(v)) among N (v)
       else
          v broadcasts P (s, v, ø, N (v)) among N (v)
       endif
    elseif ( u ∈ F(v)) then
          F (v) = F (v) - {u}
       if ( F(v) == ø ) then
          R_timer = 0
       endif
    else
       record packet
    endif
end
```

Table 5: When R_timer for a node v expires

```
begin
    if ( F(v) != ø) ∧ ( C_v(p) < RT_max ) then
      if ( atleast one node in F(v) is in 1-hop list) then
         C_v(p) = C_v(p) + 1
         R_timer = T_wait
         v broadcasts P (s, v, F(v), N(v)) among N(v)
      endif
    endif
end
```

Table 6: Simulation Scenario

Parameter	Value
Simulator	*ns*-2 (version 2.31)
Network area	$900{\times}900m^2$
Transmission range	$250m$
MAC layer	IEEE 802.11
Data packet size	64 bytes
Bandwidth	2Mb/s
Simulation time	$100s$

C_v(p) and sends the packet P again. The packet is not sent if the counter value exceeds RT_{max}. R_timer is scheduled when the packet is sent. This process continues until node v receives response from all its forwarding nodes or maximum number of retries is reached. Table 5 shows the actions taken by a node when R_timer expires.

5 Performance Evaluation

5.1 Simulation Setup

In order to analyze the performance of the proposed algorithm, the simulation was run under the $ns-2$. The simulation parameters are listed in Table 6. The network area is confined within $900{\times}900m^2$. Each node in the network has a constant transmission range of $250m$. We use a *two-ray ground reflection model* as the radio propagation model. The MAC layer scheme follows the IEEE 802.11 MAC specification. We use the broadcast mode with no RTS/CTS/ACK mechanisms for all message transmissions, including HELLO, DATA and ACK messages. The movement pattern of each node follows the *random way point*. Each node moves to a randomly selected destination with a constant speed between 0 and the maximum speed V_{max}. When it reaches the destination, it stays there for a random period T_s and starts moving to a new destination. The pause time T_s is always 0 in our simulation.

The network traffic load also affects the performance of the protocol. We change the value of Constant Packet Rate (CPR packet per second) while each packet has a constant length of 64 bytes. A node may fail to receive a message because of a transmission error, a transmission collision, or the node's out-of-range movement. After sending a message, a node waits for a period of time,

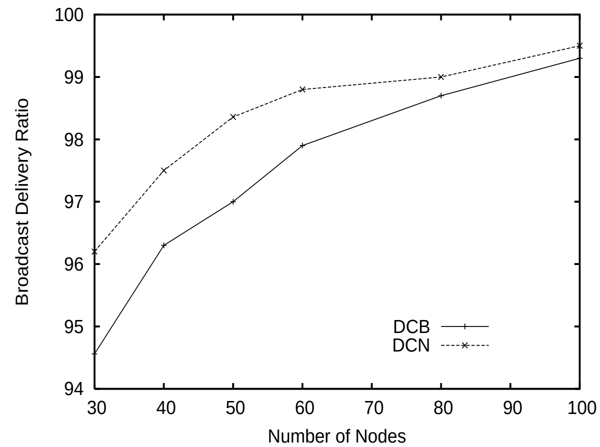

Figure 3: Sensitivity to Network Size: Delivery Ratio

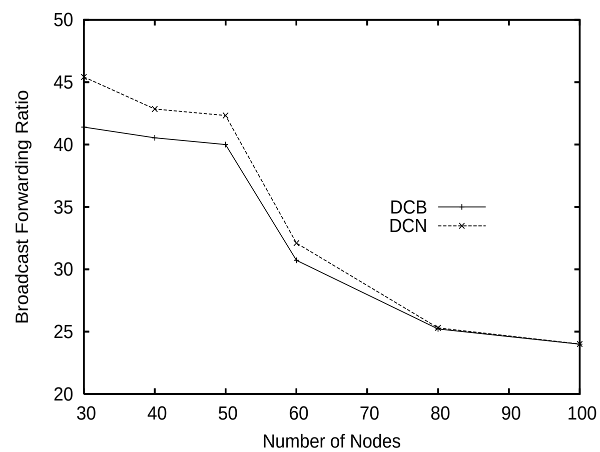

Figure 4: Sensitivity to Network Size: Forwarding Ratio

T_{wait}, and sends the message again until it reaches the maximum value R. Each simulation was run for 100 seconds. In order to avoid the initialization bias of the system state on the broadcast operation, we first make all nodes move around within the area for 30 seconds so that they can thoroughly exchange HELLO messages to build up 1-hop and 2-hop neighbor sets. Then, some randomly selected nodes start to send broadcast packets. This procedure lasts for 100 seconds. To make sure all the broadcast packets propagate throughout the network, the simulation lasts for another 10 seconds after the last broadcast process has been sent. We compare the performance of the DCN and double covered broadcast algorithm through simulation to see the advantages of DCN.

5.2 Affected parameters

We consider the following parameters that affect the performance of the broadcast:

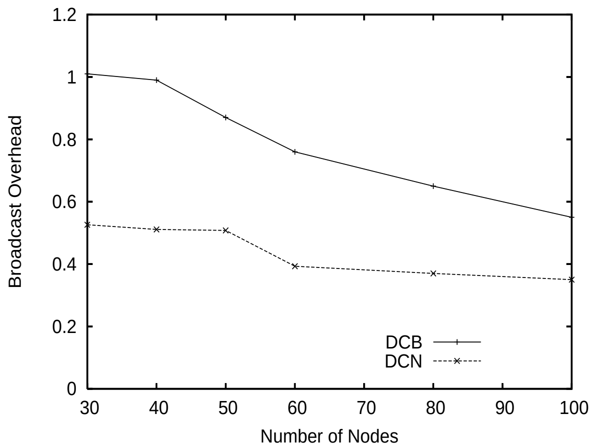

Figure 5: Sensitivity to Network Size: Overhead

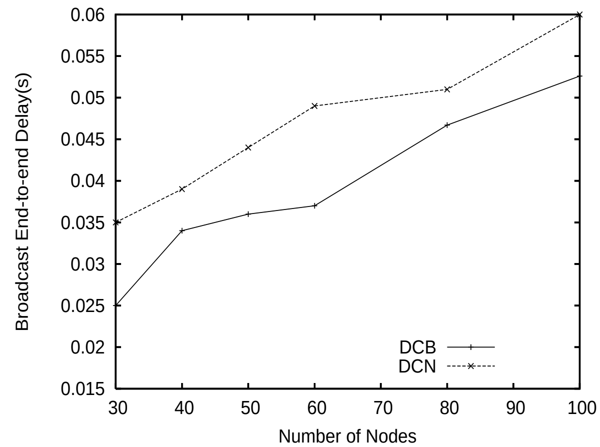

Figure 6: Sensitivity to Network Size: End-to-end Delay

(i) *Network size* (n). The number of nodes in a network determines the density of the network. A dense network causes more collision and contention.

(ii) *Transmission error rate* (P_{err}). The physical radio channel is affected by many environmental parameters. Therefore, the Signal-to-Noise Ratio (SNR) at the receiver may be below the threshold even though the receiver is in the transmission range of the sender. This affect can be estimated as transmission error rate P_{err}, which specifies a simple transmission error model in which messages may have been lost in the physical wireless channel.

(iii) *Mobility of node* (V_{max}). The mobility of the node affects the performance of the broadcast operation. The faster the node moves, the higher is the possibility of the node to lose the broadcast packet.

(iv) *Interval of HELLO messages* (T_{HELLO}). Since the nodes get neighbor information through HELLO messages, the hello interval determines the accuracy of one node's neighbor set. A large value of the interval causes the information of the neighbor set to be outdated quickly, misleading the forwarding node's broadcast decision. But, increasing the frequency of the interval also increases the overhead causing network congestion. Sending HELLO messages too frequently is similar to a flooding operation.

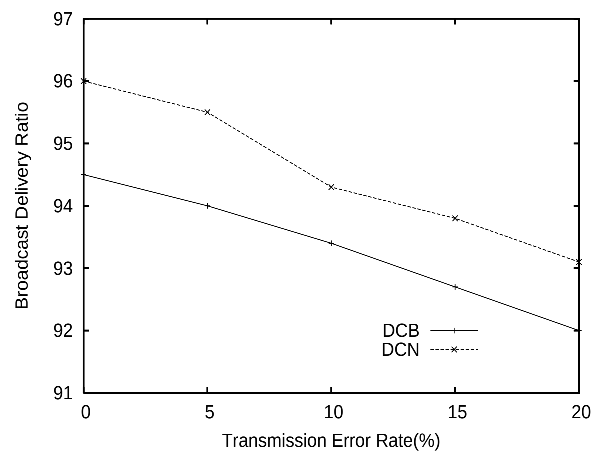

Figure 7: Sensitivity to Transmission error rate of the network: Delivery Ratio

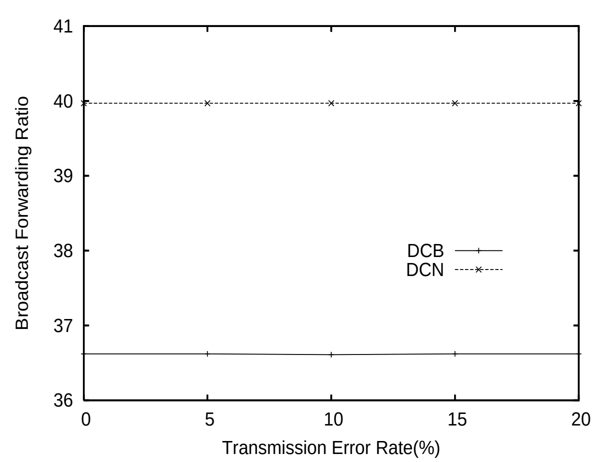

Figure 8: Sensitivity to Transmission error rate of the network: Forwarding Ratio

5.3 Results and Analysis

5.3.1 Sensitivity to network size

Figures 3, 4, 5 and 6 shows the scenario that the network has low mobility where V_{max} is 1 meter per second (m/s), and low transmission error rate ($P_{err} = 1\%$). The data traffic load CPR is 10 packets per second (pkt/s), the hello interval T_{HELLO} is 1 second (s), and the waiting time T_{wait} is 50 milliseconds (ms). We identify the effect of network size n to each metric. The network under this environment can be considered a static error free network. Most of the packet losses come from transmission collisions.

Figure 3 shows the delivery ratio. Both the algorithms have good delivery ratio (>90%). The delivery ratio of the DCN is greater than the delivery ratio of the DCB. The delivery ratio of both the algorithms is high when the network is dense. When the size of the network is small ($n = 30$), the network may sometimes disconnect, which leads to a lower delivery ratio. In sparse networks, the difference between the delivery ratio of DCN and DCB is more. But as network becomes dense, the difference between the delivery ratio of DCN and DCB reduces. This is because in sparse networks, chances of a node missing a packet is more compared to dense networks. NACK messages in DCN help the nodes to receive the missed packet. Figure 4 shows the Broadcast forwarding ratio. The DCB has less number of forwarding nodes but the difference between the forwarding ratio of DCB and DCN decreases as n increases. This is because, the delivery ratio of DCN is more compared to DCB in sparse networks. Figure 5 shows the broadcast overhead. DCN has lower broadcast overhead compared to the DCB because hello messages are piggybacked on data messages. Because of negative acknowledgements, the broadcast end-to-end delay for DCN is slightly more as shown in Figure 6.

5.3.2 Sensitivity to Transmission Error Rate

Figures 7, 8, 9 and 10 shows the performance of the algorithms under different transmission error rates. In this case $n = 30$, $V_{max} = 1m/s$, CPR $= 10pkt/s$, $T_{HELLO} = 1s$, and $T_{wait} = 50ms$. We change the transmission error rate P_{err} from 1 percent to 20 percent to see its effect on each metric.

In Figure 7, we see that the delivery ratio is affected by P_{err}. When P_{err} increases, the delivery ratio drops for both the algorithms. But, the delivery ratio of DCN is much better than DCB when P_{err} increases because of

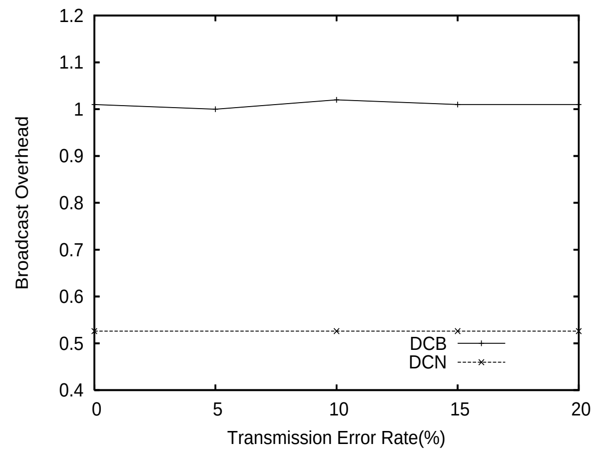

Figure 9: Sensitivity to Transmission error rate of the network:Overhead

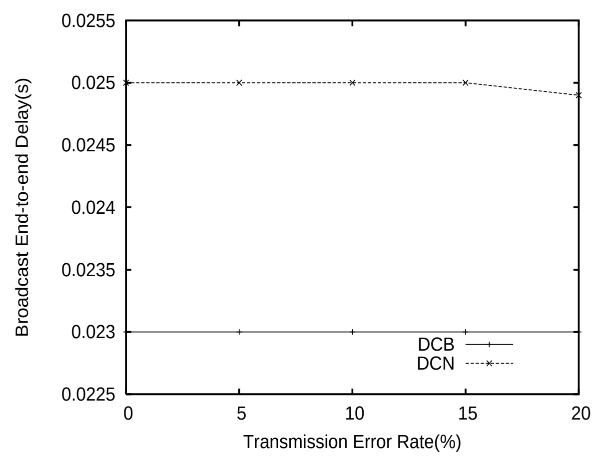

Figure 10: Sensitivity to Transmission error rate of the network: End-to-end Delay

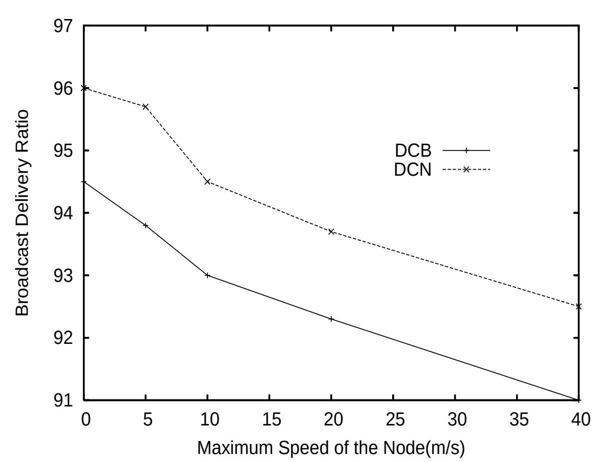

Figure 11: Sensitivity to mobility of the node: Delivery Ratio

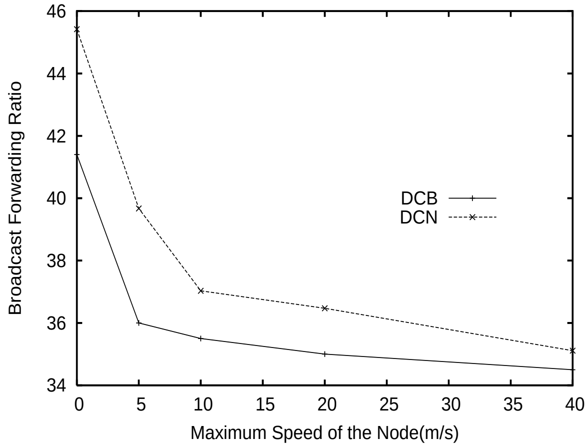

Figure 12: Sensitivity to mobility of the node: Forwarding Ratio

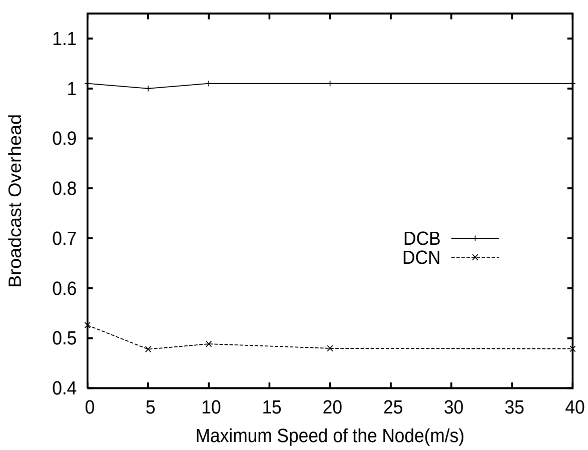

Figure 13: Sensitivity to mobility of the node: Overhead

NACK messages. It is observed from Figure 8 that the forwarding ratio of DCN is much higher than the DCB because of higher delivery ratio. It is observed from Figure 9, that the overhead of DCN is much less than the DCB because Hello messages are piggybacked into data messages. From Figures 8, 9 and 10 we can infer that forwarding ratio, broadcast end-to-end delay and overhead are not sensitive to the transmission error rate.

5.3.3 Sensitivity to Mobility of the node

Figures 11, 12, 13 and 14 shows the effect of the node's mobility on the performance of broadcast operation. In this case $n = 30$, $P_{error} = 1\%$, CPR $= 10pkt/s$, T_{HELLO} $= 1s$, and $T_{wait} = 50ms$. We change the maximum speed of each node V_{max} from 1 to $40m/s$ to show the effect of the node's mobility to each metric.

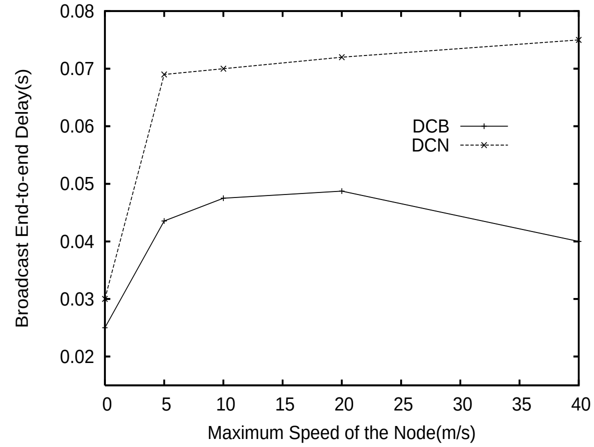

Figure 14: Sensitivity to mobility of the node: end-to-end Delay

Figure 11 shows the broadcast delivery ratio of the algorithms. The delivery ratios of DCB and DCN drops as the node's mobility increases. The chance of a node missing a packet is more as mobility of nodes increases. In this scenario, reliability is achieved in DCN by using NACK messages. Therefore the delivery ratio of DCN is more compared to DCB. The difference between the delivery ratio of DCN and DCB is constant because the mobility of the node has same effect on both the algorithms. Figure 12 shows the forwarding ratio. In sparse network, forwarding ratio is dependent on delivery ratio. Therefore as delivery ratio decreases, forwarding ratio also decreases. Figures 13 and 14 show the broadcast overhead and end-to-end delay. Mobility affects these metrics slightly. The delay of DCN has slightly increased because of negative acknowledgements. The overhead of DCN has decreased compared to DCB because of piggybacking neighbor information on data messages.

5.3.4 Sensitivity to Hello Interval

In order to investigate the effect of the hello interval on the performance of the DCN, we set the hello interval T_{HELLO} at 1 and $5s$. In this case $n = 30$, $P_{error} = 1\%$, CPR $= 10pkt/s$, and $T_{wait} = 50ms$. V_{max} ranges from 1 to $40m/s$.

In Figure 15, the delivery ratio decreases as the mobility of the node increases, especially when $T_{HELLO} = 5s$ because, as the node's mobility increases, the neighbor set information of each node is outdated quickly. Accurate neighbor information is necessary to determine

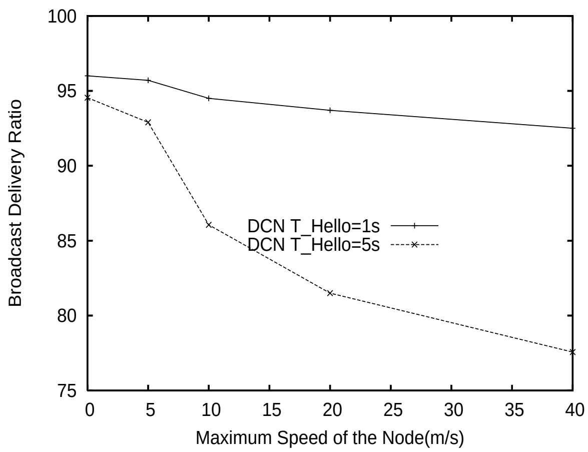

Figure 15: Sensitivity to Hello interval: Delivery Ratio

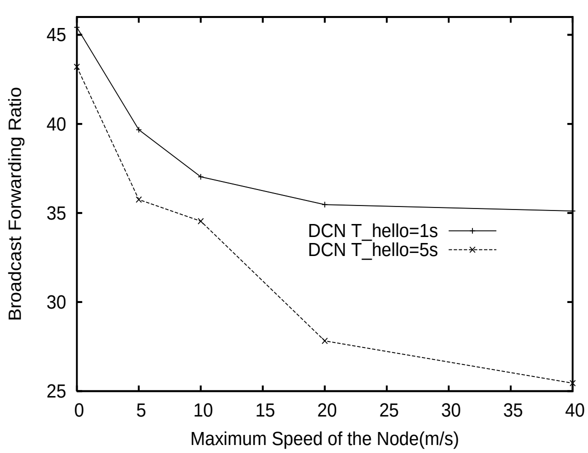

Figure 16: Sensitivity to Hello interval:Forwarding Ratio

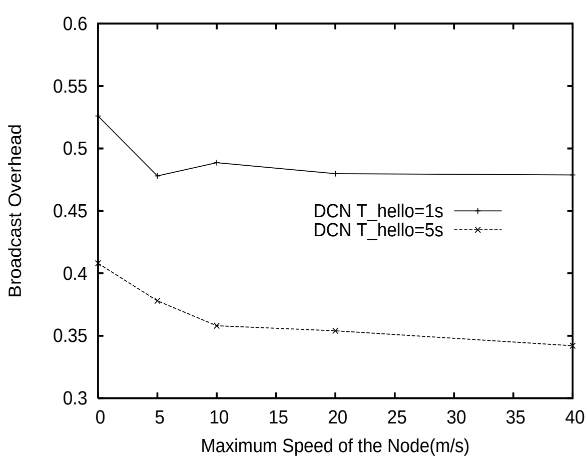

Figure 17: Sensitivity to Hello interval: Overhead

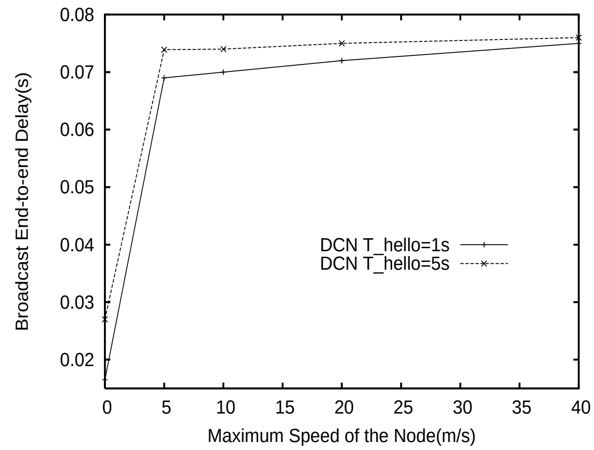

Figure 18: Sensitivity to Hello interval: End-to-end Delay

forwarding nodes which help to achieve high delivery ratio. The short hello interval causes the neighbor information to be kept more accurate in the dynamic network environment. Forwarding ratio decreases as delivery ratio decreases as shown in Figure 16. Simulation results show that infrequent updation of neighbor information results in outdated data, while exchanging the HELLO messages too frequently generates large overhead which is observed from Figure 17. From Figure 18, it is observed that, when the hello interval is low, hello messages are exchanged frequently resulting in increased end-to-end delay.

6 Conclusions and Future Work

In this paper, we propose Dual Covered broadcast with Negative Acknowledgements. Reliability and packet delivery ratio are of key concern in Mobile Ad hoc Networks. This algorithm takes advantage of redundant data packets in the network to achieve high delivery ratio. The proposed algorithm achieves reliability avoiding problems in broadcasting i.e., ACK implosion and Broadcast storm. Piggybacking neighbor information on data packets reduces broadcast overhead and delay as compared to Double Covered Broadcast algorithm. The use of NACK messages ensures reliable broadcasting. Some of the advantages of this algorithm are: only the forwarding nodes transmit the packet so that the broadcast collision and congestion are reduced; the retransmissions of forwarding nodes treated as ACKs by sender, avoids the ACK implosion problem and the loss of broadcast packets can be recovered in a local region.

Simulation results shows that the delivery ratio for dense network is high compared to sparse network. NACK messages have improved the delivery ratio for sparse network compared to Double covered broadcast algorithm which does not use NACK messages. The future work involves having two transmission ranges: the lower range for Hello messages and higher range for data, to overcome loss of packets due to random dynamic mobility of nodes.

References

[1] Min Sheng, Jiandong Li, Yan Shi, "Relative Degree Adaptive Broadcast Algorithm for Adhoc Networks," *IEEE Transactions on Broadcasting,* vol. 51, no. 2, June 2005.

[2] Mansoor Mohsin, David Cavin, Yoav Sasson, Ravi Prakash, Andre Schiper, "Reliable Broadcast in Wireless Mobile Adhoc Networks," *Proceedings of Eleventh International Conference on Sciences,* 2006.

[3] Jie Wu, Wei Lou, "Efficient Broadcast with Forward Node Set in Clustered MANETs," *Proceedings of Eleventh International Conference on Computer Communications and Networks,* October 2002.

[4] Shen Jun, Yang Kun, Cao Yuanda, Su Dengfu, "RLABP: A Receiver-Deciding Location-Aided Broadcast Protocol for Mobile Adhoc Networks," *IMACS Multiconference on Computational Engineering in Systems Applications,* vol. 2, October 2006.

[5] Fei Dai, Jie Wu, "Broadcasting in Ad Hoc Networks based on Self Pruning," *Proceedings of IEEE INFOCOM,* 2003.

[6] Song Yean Cho, Jin Hyun Sin, Byung In Mun, "Reliable Broadcast Scheme Initiated by Receiver in AHNs," *Proceedings of 28th Annual IEEE International Conference On Local Computer Networks,* 2003.

[7] Wei Lou, Jie Wu, "On Reducing Broadcast Redundancy in Adhoc Wireless Networks," *Proceedings of IEEE Transactions on Mobile Computing,* vol. 1, April-June 2002.

[8] Ivan Stojmenovic, Mahtab Seddigh, Jovisa Zunic, "Dominating Sets and Neighbor Elimination-Based Broadcasting Algorithms in Wireless Networks," *Proceedings of IEEE Transactions on Parallel and Distributed Systems,* vol. 13, no. 1, January 2002.

[9] A. Qayyum, L. Viennot, A. Laouiti, "Multipoint Relaying for Flooding Broadcast Message in Mobile Wireless Networks," *Proceedings of 35th Hawaii International Conference on Systems Sciences (HICSS-35),* January 2002.

[10] D. Towsley, J. Kurose, S. Pingali, "A Comparison of Sender-Initiated and Receiver-Initiated Reliable Multicast Protocols," *IEEE Journal on Selected Areas in Communications,* vol. 15, no. 3, April 1997.

[11] P. Bhagwat, P. Mistae, S. Tripathi, "Effect of Topology on Performance of Reliable Multicast Communication," *Proceedings of IEEE INFO-COM 94,* June 1996.

[12] J. C. Lin, S. Paul, "RMTP A Reliable Multicast Transport Protocol," *Proceedings of IEEE INFOCOM 96,* April 1996.

[13] A. Erramilli, R. P. Singh, "A Reliable and Efficient Multicast Protocols for Broadband Broadcast Networks," *Proceedings of ACM SIGCOMM 88,* pp. 343-352, August 1988.

[14] S. Floyd, V. Jacobson, S. McCanne, C. G. Liu, L. Zhang, "A Reliable Multicast Framework for Light Weight Sessions and Application Level Framing," *Proceedings of ACM SIGCOMM 95,* pp. 342-356, August 1995.

[15] S-T. Shue, Y. Tsai, J. Chen, "A Highly Reliable Broadcast Scheme for IEEE 802.11 Multihop Adhoc Networks," *Proceedings of 2002 IEEE International Conference on Communications (ICC'2002),* vol. 1, pp. 610-615, April, May 2002.

[16] P. M. Merlin, A. Segall, "A Failsafe Distributed Routing Protocol," *IEEE Transactions on Communications,* vol. 27, pp. 1280-1288, 1979.

[17] J. J. Garcia-Luna-Aceves, Y. Zhang, "Reliable Broadcasting for Dynamic Networks," *Proceedings of 1996 IEEE International Conference on Communications (ICC'1996),* vol. 3, pp. 1630-1634, June 1996,

[18] S. Alagar, S. Venkatesan, J. Cleveland, "Reliable Broadcasting for Mobile Wireless Networks," *Proceedings of Military Communications Conference (MILCOM'1995),* pp. 236-240, November 1995.

[19] S. Pleisch, M. Balakrishnan, K. Birman, R. Renesse, "MISTRAL: Efficient Flooding in Mobile Adhoc Networks," *Proceedings of ACM MobiHoc'06,* May 2006.

[20] E. Pagani, G. P. Rossi, "Providing Reliable and Fault Tolerant Broadcast Delivery in Mobile Adhoc Networks," *Proceedings of Mobile Networks and Applications,* vol. 4, pp. 175-192, November 1999.

[21] Wei Lou, Jie Wu, "Towards Broadcast Reliability in Mobile Adhoc Networks with Double Coverage," *IEEE Transactions on Mobile Computing,* vol. 6, no. 2, February 2007.