Int. J. of Advanced Networking and Applications
Volume: 01 Issue: 03 Page: 147-153 (2009)

Reliability in MANET's Using Enhanced Double Coverage Broadcasting (EDCB)

R.Balakrishna
Dept of CSE, Don Bosco Institute of Technology, Bangalore.
Research Scholar, Sri Krishna Devaraya University
Email: rayankibala@yahoo.com

U.Rajeswara Rao, G.A.Ramachandra
Dept of Mathematics and CS&T,
Sri Krishna Devaraya University, Anatapur, Andhra Pradesh.

---ABSTRACT---
The broadcast operation, as a fundamental service in mobile adhoc networks (MANETs), is prone to the broadcast storm problem if forwarding nodes are not carefully designated. The objective of reducing broadcast redundancy while providing high delivery ratio under high transmission error rate is a major challenge in MANETs. This paper proposes a simple broadcast algorithm, called enhanced double-covered broadcast (EDCB), which takes advantage of broadcast redundancy to improve the delivery ratio in an environment that has rather high transmission error rate.

Key words: Reliability, enhanced double coverage, MANETS, Performance evaluation, forwarding node

Date of Submission: December 03, 2009 Accepted: January 10, 2010

1. INTRODUCTION

A mobile adhoc network (MANET) enables wireless communications between participating mobile nodes without the assistance of any base station. The two nodes which are not in each other transmission range calls for a supporting intermediate node that relays the message and sets up communication. Further the broadcast operation place important fundamental role in the MANETs system by the way of radio transmission. With an added advantage of a single node transmitting packets, to the neighbors who can receive the message. But the MANETs system suffers from high transmission error rate owing to the high transmission contention and congestion. Hence providing higher reliability for broadcasting operation under dynamic MANET[16] system becomes an important concern and it posses a major challenge. And may be tackled by selecting an ideal error free fully coveraged environment with high transmission delivery ratio, single forwarding node which is entirely different from the existing algorithms that are based on probability.

In that direction the acknowledgment messages (2ACKs)[16], used to ensure broadcast delivery ratio and E2ACK[17,16] fulfilling the requirements of receivers to send ACKs in response to the reception of a packet may becomes another bottleneck which rises to channel congestion and packet collision, are has to be effectively addressed. Hence the present paper aims at ad hoc networks using double coverage for improved broadcasting reliability.

2. Enhanced DOUBLE COVERAGE BROADCAST (EDCB) – A NOVEL PROPOSAL

With the goal of reducing the number of forwarding nodes without sacrificing the broadcast delivery ratio, here with it is proposed with a simple broadcast algorithm designated as called *double-coverage broadcast* with feature of broad casting redundancy and higher delivery ratio with lowered high transmission error rate.

G = (V, E): a unit disk graph as in Fig 1
Where

V: Node set (set of wireless mobile nodes)
E: Edge set (set of bi-directional links between neighboring nodes)

Two nodes are considered as neighbors if and only if their geographic distance is less than the transmission range *r*. In a localized broadcast protocol, a node *v* is equipped with a *k*-hop subgraph *Gk* (*v*) for a small *k*, such as *k* = 2 or 3. *Gk* (*v*), induced from *k*-hop information of *v*, is [*Nk* (*v*), *Ek* (*v*)]. *Nk* (*v*) denotes the *k*-hop neighbor set of node *v* which includes all nodes within *k* hops from *v* (and also includes *v* itself). *Hk* (*v*) denotes the *k*-hop node set of *v* which includes all nodes that are exactly *k* hops away from *v*; that is, *N*0(*v*) = *H*0(*v*) = *fvg*, *Nk* (*v*) = *Nk*¡1(*v*) [*Hk*(*v*), *Hk*(*v*) = *Nk*(*v*) / *Nk*¡1(*v*), for *k*=1. For convenience, 1-hop neighbor set *N*1 (*v*) and 1-hop node set *H*1 (*v*) are represented as *N* (*v*) and *H* (*v*), respectively. *Ek* (*v*) denotes the set of links between *Nk* (*v*), excluding those links between *Hk* (*v*). That is, *Ek* (*v*) = *Nk*1 (*v*) £ *Nk* (*v*). For example, if *v* has 1-hop neighbor information, then it knows all its neighbors, but not the links between these neighbors. If *V* is a node set, *N* (*V*) is the union of the neighbor sets of every node in *V*, that is, *N* (*V*) = *[8w2V N* (*w*)].

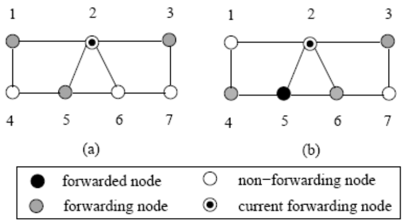


Fig1: Disk graph of MANETS

2.1 Algorithms

2.1.1 Dynamic Neighbor-designating Broadcast Algorithm

1. Let u be the senders broadcast packet, let it designate some neighboring node as its forwarding node set F(u) to cover its 2 hop node set H2(u) thus u sends the packet together with F(u).
2. As the node v receives the packet form u for the first time, if v is not designated as a forwarding node by u, it does nothing; otherwise it becomes a new sender and goes to step 1 above and is as shown in the Fig 2.

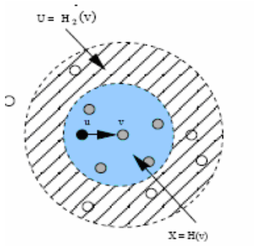


Fig 2. Multiple relays (MPR)

The algorithms of the class of dynamic neighbor-designating broadcast algorithms adopting greedy strategy so as to designate minimum number of forwarding nodes are selected so that other neighbors can take the non-forwarding status.

In this proposal, multipoint relays [8, 9] are adopted selected as the forwarding nodes to propagate link state messages. The MPRs are selected from 1-hop neighbors to cover 2-hop neighbors. Forwarded nodes are not considered for a node to select its MPRs and, therefore, the entire set of 2-hop neighbors must be covered (Figure 1(a)). Specifically, *v* selects its forwarding node set *F* from all candidate neighbors *X* = *H* (*v*) = *N* (*v*) ¡ *fvg* to cover its uncovered 2-hop neighbors *U* = *H*2 (*v*) = N2 (*v*) ¡ *N* (*v*) with a simple greedy algorithm used in the set coverage problem [10]. These forwarding nodes set selection process (FNSSP) are detailed as follows.

2.1.2: Forwarding Node Set Selection Process (FNSSP) (for node *v*)

1. Initially, *X* = *H* (*v*), *U* = *H*2 (*v*), and *F* = *Á*.
2. Find *w* (in *X*) with the maximum effective neighbor degree *degree* (*w*) = *jN*(*w*) \ *Uj*.
3 .*F* = *F [fwg*, *U* = *U ¡ N*(*w*), and *X* = *X ¡ fwg*.
4. Repeats step 2 and 3 above until *U* becomes empty as illustrated in Fig 3.

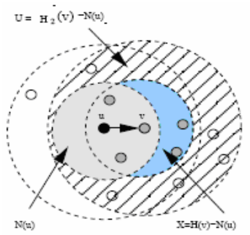


Fig 3: Dominant Pruning

2.1.3 Enhanced Double-Covered Broadcast Algorithm

Initiation

Broad casting network requires a packet to receive by all nodes in the network. But transmission interference and the movement of the nodes may cause some nodes to lose the broadcast packet. The redundancy of the broadcast packet can bring more opportunities for a node to receive the packet successfully. Moreover, if the sender can retransmit the packet, the number of nodes that receive the broadcast packet is also increased.

The proposed enhanced double-covered broadcast (EDCB) algorithm [1,6] works as follows
When a sender broadcasts a packet, it selects a subset of 1-hop neighbors as its forwarding nodes to forward the packet based on a greedy approach. The selected forwarding nodes satisfy the following:

1. They cover all the sender's 2-hop neighbors
2. The sender's 1-hop neighbors are either forwarding nodes, or non-forwarding nodes but covered by at least two forwarding nodes

After receiving a new broadcast packet, each forwarding node records the packet, computes it's forwarding nodes and rebroadcasts the packet as a new sender. The retransmissions of the forwarding nodes are overhead by the sender as the acknowledgement of the reception of the packet. The non-forwarding 1-hop neighbors of the sender do not acknowledge the receipt of the broadcast. The sender waits for a predefined duration to overhead the rebroadcast from its forwarding nodes. If the sender fails to detect all its forwarding nodes retransmitting during this duration[7,9], it assumes that a transmission failure has occurred for this broadcast. The sender then resends the packet until all the forwarding nodes' retransmissions are detected or the maximum number of retries is reached. The sender may miss a retransmission from a forwarding node, and therefore resends the packet. When the forwarding node receives a duplicated broadcast packet, it sends an ACK to acknowledge the sender.

The EDCB algorithm selects a set of forwarding nodes that form a virtual backbone of the network. The forwarding nodes are selected in such a way that they balance the average retransmission redundancy for the delivery of a broadcast packet throughout the entire network. The scheme avoids the broadcast storm problem: since only the forwarding nodes transmit the packet, the broadcast collision and congestion are reduced. This scheme also avoids the ACK implosion problem: the retransmissions of forwarding nodes are also used as the ACKs to the sender so that no extra ACKs are needed. The failure of overhearing forwarding nodes' relays will trigger the sender to retransmit the packet, so that the packet loss can be recovered in a local region. Each non-forwarding node is covered by at least two forwarding neighbors so that it can tolerate a single transmission error and its chance to receive the broadcast packet successfully is greatly increased even in a high transmission error rate environment. Moreover, the algorithm does not suffer the disadvantage of the receiver-initiated approach that needs a much longer delay to detect a missed packet.

Forwarding Node Set Selection Process

Assuming that each node *v* knows its 2-hop sub graph *G*2 (*v*) = (*N*2 (*v*); *E*2 (*v*)). A forwarding node *v* uses the FNSSP-DC (Algorithm 3) to determine its forwarding node set *F*(*v*): *v* uses the FNSSP algorithm (Algorithm 2) to find *F*(*v*) in *H*(*v*) to cover *N*2(*v*) (*fvg* (Figure 2(a)). Unlike the MPR algorithm [8] where only nodes in *H*2(*v*) need to be covered by forwarding node set *F*(*v*), the FNSSP-DC algorithm guarantees that *v*'s 2-hop neighbor set *N*2(*v*) (excluding *v* itself) is completely covered by *v*'s forwarding node set *F*(*v*). Since *v* also transmits the packet to cover *H* (*v*), any non-forwarding node in *H* (*v*) is covered twice and is shown in Fig 4.

1. Each node *v* computes *X* = *H* (*v*) and *U* = *N*2 (*v*) *fvg*.
2. Node *v* uses the FNSSP to find *F*(*v*) in *X* to cover *U*.

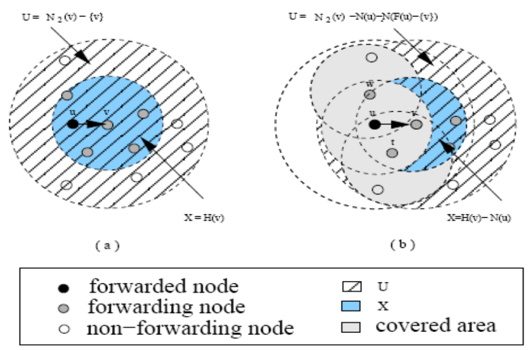


Fig 4: Illustrations of the forwarding node set selection process of the EDCB algorithms.

The source of a broadcast operation uses the FNSSP-DC algorithm to determine its forwarding node set. Other forwarding nodes consider the impact of the sender of the broadcast packet. If *v* is a designated forwarding node of *u*, that is, *v* receives a new packet from *u* and *v* finds itself in *F*(*u*), *v* uses the FNSSP-EDC algorithm (Algorithm 4) to determine its forwarding node set (Figure 2(b)): *v* finds *F*(*v*) in *H*(*v*)¡*N*(*u*) to cover *N*2(*v*)¡*N*(*u*)¡*N*(*F*(*u*)¡*fvg*). The goal of FNSSP-EDC is to cover all those nodes in the 2-hop neighborhood of *v*, excluding those that have been already covered by *u* and those that will be covered by some other forwarding nodes of *u*.

2.1.4 Forwarding Node Set Selection Process - Enhanced Double Coverage (FNSSP-EDC)

Referring to Fig 1, a sample network where node 2 uses the FNSSP-EDC to select its forwarding nodes

1. Each node *v* sets *X* = *H*(*v*) ¡ *N*(*u*) and *U* = *N*2(*v*) *N*(*u*) / *N*(*F*(*u*) *fvg*).
2. Node *v* uses the FNSSP to find *F* (*v*) in *X* to cover *U*.

2.1.5 The Enhanced Double-Covered Broadcast (EDCB) Algorithm

Notations

F (v): the forwarding node set of node v.
U (v): the uncovered 2-hop neighbour set of node v.
X (v): the selectable 1-hop neighbor set of node v.
P(u), F(u)): unique broadcast packet P forwarded by node v that attaches vs. forwarding node set F (v).
T_{wait}: the predefined duration of a timer for a node to overhear the retransmission of its forwarding nodes.
R: the maximum number of retries for a node.

Proposed Algorithm

1. When source *s* wants to broadcast *P*, it uses the FNSSP-DC to find *F*(*s*) and broadcasts *P*(*s; F*(*s*)).
2. When node *v* receives *P*(*u; F*(*u*)) from *u*,
   v records *P*(*u; F*(*u*)).
   v updates *X*(*v*) = *X*(*v*) ¡ *N*(*u*) and *U*(*v*) = *U*(*v*) ¡ *N*(*u*) ¡ *N*(*F*(*u*) ¡ *fvg*).
   if *v* 2 *F*(*u*) then if the packet has not been received before then *v* uses the FNSSP-EDC to find *F*(*v*) that covers *U*(*v*) and broadcasts *P*(*v; F*(*v*)).
   Else, *v* sends an ACK to *u* to confirm the reception of *P* and drops the packet.
   End if else *v* drops the packet.
   end if
3. When node *u* has sent the packet, it starts a timer T_{wait} and overhears the channel. After T_{wait} is expired, if *u* does not overhear all nodes in *F*(*u*) to resend *P* or to send ACKs, *u* retransmits *P* until the maximal number of retries *R* is reached.

The above concept is supported by the following theorem, *"Given a connected network, the EDCB algorithm works correctly based on the assumption that broadcasting through this network is an atomic operation"*

Proof: We prove Theorem 2 by contradiction. Assume that the network is not fully covered when broadcasting a packet with the EDCB algorithm, that is, we can find at least one node *d* such that *d* does not receive the broadcast packet from the source *s*. In Figure 5, the set *C* inside the circle represents the covered node set, *C* represents the uncovered node set. Therefore, *s 2 C* and *d 2 C*. Since the network is connected, there exists a path from *s* to *d*. Suppose node *x* is the uncovered node that is closest to *s* on the path, and *v* is the predecessor of *x* on the path. Based on the assumption, *v* has received the broadcast packet, say *v* has received the packet from node *u* for the first time. Because *x 2 N*2(*u*), Theorem 1 guarantees that *u* covers *x*. This contradicts the assumption. Therefore, the EDCB algorithm guarantees the network is fully covered.

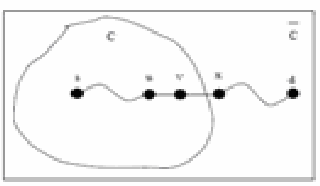


Fig 5. Illustration of the Proof of the Theorem

Working of the proposed EDCB Algorithm

1. When a node *s* starts a broadcast process, *s* uses the FNSSP-DC algorithm to select its forwarding node set *F*(*s*), and broadcasts the packet *P* together with *F*(*s*).

2. When a node *v* receives *P* from an upstream sender *u*, it records *P*. *v* also updates its *X*(*v*) = *X*(*v*)¡*N*(*u*) and *U*(*v*) = *U*(*v*)¡*N*(*u*)¡*N*(*F*(*u*)¡*fvg*). Note that *X*(*v*) and *U*(*v*) are initialized to *H*(*v*) and *H*2(*v*). Then, *v* checks whether it is a designated forwarding node of *u*. If not, *v* drops the packet and stops the process; otherwise, *v* further checks whether *P* is ever received. If *P* is a new packet for *v*, *v* uses the FNSSP-EDC algorithm to compute its forwarding nodes *F*(*v*) and sends *P* with *F*(*v*). If *v* has already received *P* from another node, *v* will not forward *P*, but send an ACK to *u* to confirm the reception so that *u* will not retransmit the same packet at a later time.

3. When the sender *u* broadcasts *P*, it waits for a predefined duration T_{wait} to overhead the retransmission of its forwarding nodes. If *u* overhears a retransmission packet from its forwarding node *v*, *u* regards this as an ACK from *v*. *u* may receive explicit ACKs from some of its forwarding nodes to confirm the reception. If *u* does not overhear all of its forwarding nodes when the timer expires, it assumes that the transmission failure has occurred for this packet. *u* then determines a new *F*(*u*) to cover the rest of the uncovered *U*(*u*) and resends the packet until the maximal number of retries *R* is reached. The Fig 6 Shows working of EDCB

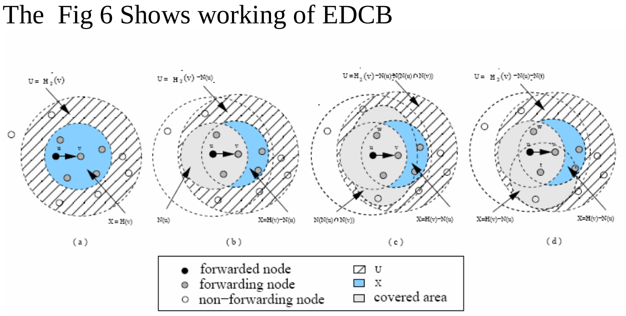


Fig 6 Working of Proposed EDCB Algorithms

3. Simulation

In order to analyze the performance of the proposed algorithm, we run the simulation under the Glomosim simulator test bed with SMTP wireless extension. The simulator parameters are listed in Table 1(a): the network area is confined within 1000×1000 m^2. Each node in the network has a constant transmission range of 350 m. We use a two-ray ground reflection model as the radio propagation model. The MAC layer scheme follows the IEEE 802.11 MAC[6,7,9] specification. We use the broadcast mode with no RTS/CTS/E2ACK mechanisms for all message transmissions, including HELLO, DATA and E2ACK[16,17] messages. Since transmission errors may occur when nodes send messages in real wireless channels, we assume a probability P for each wireless channel to successfully transmit a message. The movement pattern of each node follows the random way-point model: 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 will wait for a period of time T_{wait} and resend the message 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 1000 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 will last for another 10 seconds after the last broadcast process has been sent. We run the simulation 10 times to achieve a 90% confidence interval for the results.

Table 1. Simulator Parameters

Parameter	**Value**
Simulator	Advent
Network area	1000 X 1000
Transmission Rang	350m
MAC Layer	IEEE 802.11
Data Packet Size	64 Bit
Band width	4 Mb/s
Simulation Time	90 s
Number Trials	10
Confidence Intervals	93 %

4. Result Analysis

In this 1000x1000 network area at low mobility where V_{max} is 1 meter per second(m/s) and low transmission error rate (P_{err}= 0.8%). For data traffic load CPR at 10 packets per second (pkt/s), the hello interval HELLO is 1 second (s), and the waiting time T_{wait} is 50 millisecond (ms). the identified effect of network size is n to each metric and the network under this environment can be considered being static error-free. It is observed for the above conditions all algorithms have good delivery ratios (> 93%). Further the delivery ratio proposed EDCB-SD is greater than the other two algorithms namely DCB-ST, DCB-RE for all the ranges considered. Where as the delivery ratios of all DCB algorithms are slightly higher than AHBP-EX at dense (n=1000 network),which suggesting the EDCB-SD algorithm delivery ratio is superior benefiting from the retransmission mechanism. The fig7-10 are presented with the performance analysis of the Proposed EDCB-SD algorithm. Fig 7 shows the packet delivery ratio for the different nodes. From fig it can be observed that with increased nodes the packet delivery ratio increases. Further form the nature of the curve it may be depicted that the packet delivery ratio increases with increased number of nodes.

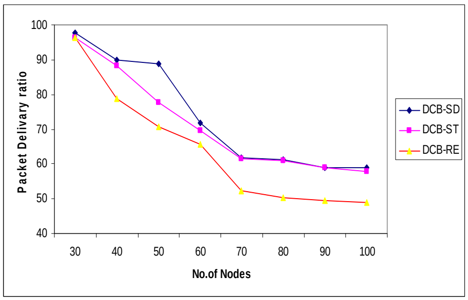


Fig 7. Variations Delivery Ratio with increased number of nodes

Fig 8 shows the Broad Casting ratio for the different nodes. From fig it can be observed that with increased nodes the broadcasting delivery ratio increases. Further from the nature of the curve it may be depicted that the broad casting delivery ratio increases with increased number of nodes.

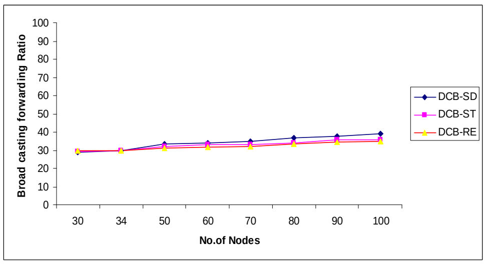


Fig 8. Variations Broad Casting Ratio with increased number of nodes

Fig 9 shows the broadcasting overhead for the different nodes. From fig it can be observed that with increased nodes the broad casting over head increases. Further from the nature of the curve it may be depicted that the broad casting overhead decreases with increased number of nodes.

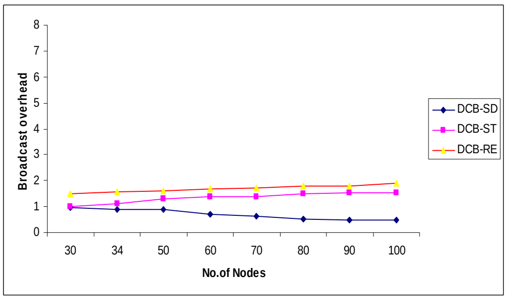


Fig 9. Variations overhead with increased number of nodes

Fig 10 shows the Broad Casting end to end delay for the different nodes. From fig it can be observed that with increased nodes the broadcasting end to end delay increases. Further from the nature of the curve it may be depicted that the broad casting delivery ratio increases with increased number of nodes.

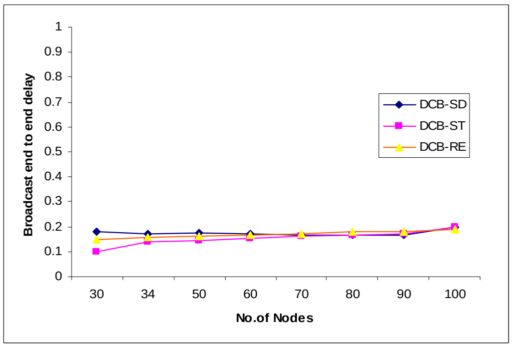


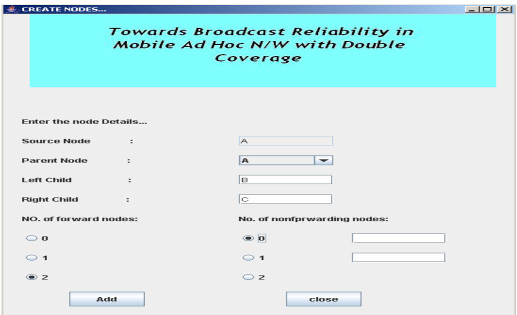
Fig 10. Variations end to end delay with increased number of nodes

From this simulation, we can see that all algorithms have high delivery ratios under the 1000x1000 that the network is almost static and transmission error-free. DCBs and AHBP-EX have comparable performance under this scenario. Also, we notice that EDCB-SD performs best among three DCB algorithms

5. Implementation

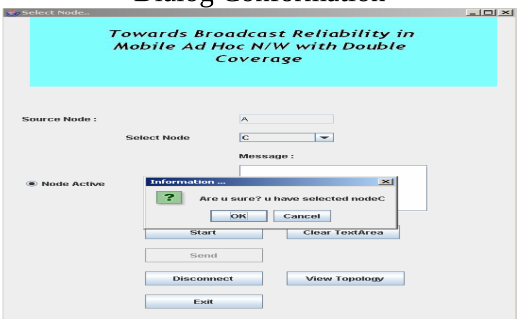
The steps followed in the implementation of proposed EDCB-SD algorithms are shown

Node creations

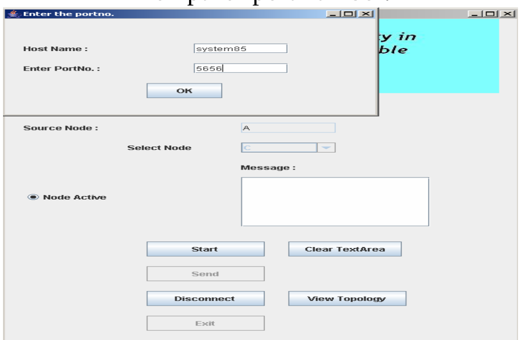


Selection of Node

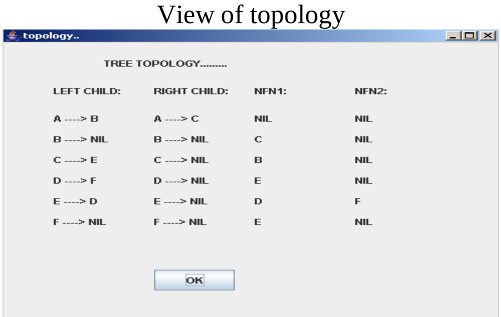
Dialog Conformation



Prompt for port number.



View of topology



6. Advantages of EDCB and Comparisons

The advantages of proposed EDCB are as follows

1. Data redundancy is avoided, so that each node is promised to receive the packet only from one node.
2.The total delay for getting acknowledgement at source is reduced since data redundancy is removed.
3. In case of node failure, the node topology is rearranged so that child nodes of failed node receive the packets

The comparison among various algorithms shown in Table 2 supports the proposed EDCB-SD algorithms:

Algorithm	Description		
	Transmit	Acknowledge	retransmit
EDCB-SD	Forwarding nodes	Forwarding nodes	Reselect
DCB-ST	Forwarding nodes	Forwarding nodes	Resend
DCB-RE	Forwarding nodes	Forwarding nodes	Recalculate

Table 2 Comparison of various algorithms

7. Conclusions

Carried at on the extensive work EDCB are concluded as follows:

Algorithm provides high delivery ratio while suppressing broadcast redundancy. This is achieved by only requiring some selected forwarding nodes among the sender's 2-hop neighbor set to forward the packet. The double-covered forwarding node set selection process provides some redundancy to increase the delivery ratio for non-forwarding nodes so that retransmissions can be remarkably suppressed when transmission errors are considered. The simulation results show that the double-covered broadcast algorithm has

- high delivery ratio,
- low forwarding ratio,
- low overhead and low end-to-end delay

for a broadcast operation under high transmission error ratio environment. From the simulation, we observe that the EDCB is sensitive to the node's mobility.

The EDCB provides full reliability for all forwarding nodes but not for non-forwarding nodes. In order to provide full reliability for all non-forwarding nodes, we can use the N-ACK mechanism such that a non-forwarding node will send a N-ACK message when the node notices a packet loss during the continuous broadcasting transmissions.

Acknowledgements

The authors are thankful for the encouragement and support received throughout this research work to Wayanamac Education Trust (WET), Bangalore. Principal Dr.K.Muralidhar, DBIT, Dr. M.S.Bhaghashekar Head R&D, Prof N.S.Nagaraj, Head, Dept of CSE, DBIT, Bangalore.

References

[1] M. Impett, M. S. Corson, and V. Park, "A receiver-oriented approach to reliable broad cast ad hoc networks,"*Proc. of Wireless Communications and Networking Conference (WCNC'2000), vol. 1, pp. 117–122, Sep. 2000.*

[2] W. Lou and J. Wu, "A reliable broadcast algorithm with selected acknowledgements in mobile ad hoc networks," *Proc. of IEEE Global Telecommunications Conference (GLOBE-COM 2003), Dec. 2005.*

[3] Z. J. Haas, J. Y. Halpern, and L. Li, "Gossip-based ad hoc routing," *Proc. of INFOCOM2002, vol. 3, pp. 1707–1716, Jun. 2004.*

[4] R. Chandra, V. Ramasubramanian, and K.P. Birman, "Anonymous gossip: Improving mul-ticast reliability in mobile ad-hoc networks," *Proc. of IEEE Int'l conf. on Distributed Com-puter System (ICDCS'2001), pp. 275–283, Apr. 2001.*

[5] J. Wu and F. Dai, "A generic distributed broadcast scheme in ad hoc wireless networks," *Proc. of ICDCS'2003, pp. 460–468, May 2003.*

[6] B. Williams and T. Camp, "Comparison of Broadcasting Techniques for Mobile Ad Hoc Networks," *Proc. of 3rd ACM Int'l Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC'2002), pp. 194-205, Jun. 2002.*

[7] W. Lou and J. Wu, "On reducing broadcast redundancy in ad hoc wireless networks," *IEEE Trans. on Mobile Computing, vol. 1, no. 2, pp. 111–123, Apr.-Jun. 2002.*

[8] S.-T. Shue, Y. Tsai, and J. Chen, "A highly reliable broadcast scheme for IEEE 802.11 multi-hop ad hoc networks," *Proc. of 2002 IEEE Int'l Conf. on Communications (ICC'2002), vol.1, pp. 610–615, Apr.-May 2002.*

[9] S. Pleisch, M. Balakrishnan, K. Birman, and R. Renesse, "MISTRAL: Efficient Flooding in Mobile Ad-hoc Networks,"*accepted to appear in Proc. of ACM MobiHoc'06, Florence,Italy, May, 2006.F. Harary and T. Haynes*, "Double domination in graphs,"*ARS Combinatoria, vol. 55, pp. 201–213, 2000.*

[10] M. Satratzemi and K. Tsouros, "Double domination algorithms in graphs," *Proc. of Hellenic European Conference on Mathematics and Informatics (HERMIS), vol. 55, pp. 201–213, 2000.*

[11] J. Harant and M. A. Henning, "On double domination in graphs," *Discussiones Mathemat-icae, Graph Theory, Vol. 25, pp. 29–34, 2005.*

[12] H. Koubaa and E. Fleury, "On the performance of double domination in ad hoc networks," *Proc. IFIP Medhoc, Jun. 2003.*

[13] C. S. Liao and G. J. Chang, "k-tuple domination in graphs," *Information Processing Letters, vol. 87, no. 1, pp. 45–50, 2003.*

[14] J. Wu and F. Dai, "Mobility management and its applications in efficient broadcasting inmobile ad hoc networks," *Proc. of IEEE INFOCOM 2004, Mar. 2004.*

[15]R.Balakrishna, Dr.U. Rajeswara Rao, Dr.N.Geethanjali "Enhaced Secure Authenticated Key Exchange Protocol for Credential Services",*in the IEEE World Congress Science and Engineering Technologies", France , Lavna. Nov 22-24,2008.*

[16] R.Balakrishna, M.Murali Mohan Reddy Dr.U.RajeswarRao,Dr.G.A.Ramachandra, "Routing Misbehavior Detection in MANET Using 2ACK", *in **IEEE** Advanced Computing Conference, Thapur University, Patala,2008.*

[17] R.Balakrishna, M.Murali Mohan Reddy, Dr.U.Rajeswar Rao, Dr.G.A.Ramachandra," DETECTION OF ROUTING MISBEHAVIOR IN MOBILE AD HOC NETWORKS USING ENHANCED 2ACK (E-2ACK)",*in **IEEE** Advanced Computing Conference in at , Thapur University, Patal,2008*.

[18]. R.Balakrishna, Dr.U.Rajeswara Rao, Dr.N.Geethanjali, "Secure Key Exchage protocol for Credential Services", *Published in Defence science Jounal in May 2009.*

[19].R.Balakrishna, Dr.U.Rajeswara Rao, Dr.N.Geethanjali "A secured authenticated key exchange protocol for credential services", *in 3rd International conference ICACCT-2008, Page 120-129,www.apiitindia.org/icacct2008.*

[20] R.Balakrishna , Dr.U. Rajeswara Rao, Dr.N.Geethanjali,"Video Conferencing on Mobile Ad-hoc Network", *in 2nd International Conference CCR2008,Page298-305.*

[21].R.Balakrishna, M.V.Panduranga Rao, Dr.K.C.Shet," Development of Scheduler for Real Time and Embedded System Domain", *in digital library at IEEE AINA-2008 International Conference, JAPAN,. The details about the conference is available at* http://www.aina-conference.org/2008/

Authors Biography

R.Balakrishna is an research scholar at Sri Krishna Devarya University, Anantapur, Andra pradesh, India. His research interests are in the field of Wireless adhoc network, Sensor network, Artificial Neural Networks, Data mining, Operating System and Security. He has published over 26 national and International Journals & Conferences various papers across India and other countries**.** He is the Life member of Indian Society for Technical Education and IAENG. His webpage can be found via http://www.balakrishnar.i8.com/

Dr. U.Rajeswar Rao obtained his PhD degree from Sri Krishna Devaraya University, Anantapur, Andhra Pradesh, India. He has been working as a Professor in the Department of Computer Science and Mathematics, Sri Krishna Devaraya University, Anantapur, Andhra Pradesh, India, since 1983. He has published over 50 papers National and International Journals & Conferences in the area of Mathematics & Computers Science.

Dr. G.A.Ramachandra obtained his PhD degree from Sri Krishna Devaraya University, Anantapur, Andhra Pradesh, India. India. He has been working as a Professor in the Department of Computer Science & Technolgy, Sri Krishnadevaraya university, Anatapur. He has published over 10 papers National and International Journals & Conferences in the area of Data mining, Ad hoc Net works, Neural Networks, Networking.