Method for Handling Collisions of Broadcast Packets due to Hidden Node Problem

Rinki Sharma
Department of Computer Engineering
M.S. Ramaiah School of Advanced Studies, Bangalore
rinki@msrsas.org

Govind R. Kadambi
Department of Computer Engineering
M.S. Ramaiah School of Advanced Studies, Bangalore
govind@msrsas.org

Mukundan K.N.
Broadcom Communication Technologies, Bangalore, India
mukundan@broadcom.com

ABSTRACT

In this paper a method of collision detection and retransmission of broadcast packets is proposed. In multi-hop wireless networks, broadcasting is an elementary part of the routing process. However, due to the hidden node problem, broadcasting often leads to collisions. In the presence of broadcast collisions, a node cannot populate its neighbor table completely, leading to missing out on knowledge of potential neighbors. The proposed method uses Collision Detection Pulse (CDP) to make other nodes in the vicinity aware of a collision. If transmitter of the broadcast packet detects a CDP right after the completion of transmission, it retransmits the broadcast packet. The performance of the proposed method is compared with that of IEEE 802.11 DCF (Distributed Coordination Function) standard. Simulation studies show that the proposed method outperforms the existing IEEE 802.11 DCF in discovering the neighbors of a particular node. With the proposed method, every node can discover all its neighbors within a maximum delay of 161ms.

Categories and Subject Descriptors

D.4.4 [**Communications Management**]: Computer-Communication Networks - Message sending

General Terms

Performance, Reliability

Keywords

Hidden node problem, broadcast collision, collision detection, neighbor discovery, collision detection pulse

1. INTRODUCTION

Ad-hoc networks are formed when multiple nodes come in range of each other and establish communication links, without the help of any infrastructure or base station. In ad-hoc networks, each node can act as the source, sink or router of information. Therefore, routing becomes an important part of communication in ad-hoc networks. Broadcast is an elementary operation of a routing protocol. Based on whether the routing protocol is proactive, reactive or hybrid in nature, it is necessary to exchange broadcast packets for establishment of routes, and their maintenance.

In the networks using proactive protocols, nodes can broadcast 'HELLO' packets to inform other nodes in the vicinity about their presence. Receivers of these 'HELLO' packets can use them for neighbor table generation. Periodic 'HELLO' packets, also known as 'Heartbeat' packets can be broadcasted for route maintenance, and to determine the status of the neighboring nodes / routers. Reactive protocols require the source nodes to broadcast Route Requests (RREQ) to find routes for a particular destination node. To support multihop communication, intermediate nodes also need to rebroadcast the RREQ received from the original source, till the RREQ reaches the desired destination. Though broadcasting is an elementary part of routing in ad-hoc networks, it leads to the problems of collision, redundancy, and contention in the network, as pointed out in [19]. Simultaneous broadcasts may lead to collision at the receiver. Sending acknowledgments for broadcasts is not suitable because all the nodes may simultaneously transmit the acknowledgements leading to further collisions in the network, and waste of bandwidth and node energy.

In this paper, a mechanism for collision detection of broadcast packets due to hidden node problem is proposed. In the proposed method, the sender of broadcast includes checksum bits at the end of every packet. Every node that receives a broadcast is made capable of detecting a corrupted packet by verifying the checksum. It must be noted that broadcast packet may get corrupted not only due to hidden terminal problem but due to interference also. In any case, the receiver must discard the corrupted packet. The receiver of the corrupted packet then transmits a Collision Detection Pulse (CDP) into the medium. All the nodes in the vicinity of transmitter of CDP detect it, including the nodes which had transmitted the broadcast packet just before detecting CDP. If transmitter of the broadcast packet detects a CDP right after the completion of transmission, it becomes aware that its transmission has got corrupted. The transmitter then increases its backoff time and retransmits the packet. This method assumes the nodes to be static.

The rest of this paper is organized as follows. In Section 2, related research for solving hidden node problem and handling broadcast collisions is reviewed. In Section 3, the proposed method for detecting broadcast collisions in the network is explained in detail. Section 4 consists of simulation results and their analysis. Section 5 summarizes the conclusions derived from the proposed work.

2. RELATED WORK

Many researchers have proposed numerous solutions in the past to solve the hidden node problem in wireless networks. Busy Tone Multiple Access (BTMA) based solutions such as Dual BTMA (DBTMA) [4] and Receiver Initiated-BTMA (RI-BTMA) [21] were proposed to solve hidden node problems in wireless networks. However, these protocols require two channels to

function, namely message channel and busy-tone channel, leading to additional hardware cost and complexity. Multiple Access Collision Avoidance (MACA) based protocols such as MACA for Wireless LANs (MACAW) [2] and MACA-By Invitation (MACA-BI) [16], Floor Acquisition Multiple Access (FAMA) [3], IEEE 802.11 DCF (Distributed Coordination Function) based on CSMA/CA (Carrier Sense Multiple Access / Collision Avoidance) use various forms of RTS-CTS exchange to solve the hidden node problem in networks using single channel. However, none of these protocols study the possibility of collisions between two RTS packets. Authors in [4, 17] briefly discuss the possibility of collision of RTS packets due to hidden node problem, but they concentrate mainly on data packet collisions.

Though a lot of work on broadcasting techniques is available in the literature, most of it discusses the schemes for rebroadcasts. The authors in [20] have noted that to avoid collisions among broadcasts in the network, procedure of exchanging RTS/CTS/Data/ACK is difficult to coordinate and bandwidth expensive. The paper points out that, clear-channel assessment alone does not prevent collisions among hidden nodes, and there is need for appropriate collision detection mechanisms to alleviate this problem. The paper states that most effective broadcasting protocols try to limit the probability of collision by limiting the number of rebroadcasts in the network. The same paper provides a scheme called 'Jitter and RAD', which randomly delays the scheduling of broadcast packets from network layer to MAC layer of the node. Therefore, by varying Random Assessment Delays (RAD) at different nodes, neighboring nodes attempt to acquire channels at different time thus avoiding collisions. The schemes to deal with broadcasting problems in MANETs have been studied in [19]. All these schemes suggest different methods to control rebroadcasts in order to handle collisions among broadcast packets. Neighborhood based methods of broadcasting have also been proposed in past. These methods exploit the neighborhood node's information to exchange broadcasts. One of such schemes is Self Pruning [6] where each node piggybacks the information about its adjacent nodes in each rebroadcasted packet. The receivers of this packet then check their own adjacent nodes. If their adjacent nodes are same as those present in the received packet then the packet is dropped. Otherwise, the packet is rebroadcasted. Apart from this there are other schemes such as Dominant Pruning [6], Scalable Broadcasting [11], Multi-Point Relay [13], Ad-hoc Broadcast Protocol [12], and Simplified Multicast Forwarding (SMF) for MANETs [7]. All these methods require a node to have information about their two-hop neighbors. As pointed out in [14], these methods require extra transmission overhead, particularly in dense MANETs.

The authors in [1,8,9] suggest the exchange of acknowledgements for broadcast at different levels in the protocols stack. But such solution may increase the amount of traffic in the network and contention among the nodes for medium access. The authors in [10] propose Efficient Reliable One-Hop Broadcasting (EROB). This paper points out that achieving reliable one hop broadcast is difficult because of collisions caused due to hidden terminal problem. The drawback of EROB is that it requires three different channels, one for data packet transmission and two for control packet transmission.

3. PROPOSED METHOD FOR HANDLING COLLISIONS OF BROADCAST PACKETS

In this paper, a scheme for handling collisions of broadcast packets is proposed. According to the proposed method, each broadcast packet consists of a 2 bytes long checksum field. When two nodes simultaneously transmit a broadcast packet into the network, the packets collide, leading to bits getting corrupted. When any node receives a corrupted packet, it finds the checksum field to be invalid. Immediately after this, without waiting for any random delay, the receiver nodes which found the broadcast to be corrupted send a Collision Detection Pulse (CDP) into the network. The CDP is a pulse with duration of 8µs. As stated in [15], a pulse/tone of duration of 5µs can be detected by the nodes in the network, when a node detects a pulse of a minimum duration of 5µs it gets to know that broadcast packet corruption has taken place. When the transmitter of a broadcast packet detects a CDP right after completing the transmission, it gets to know that the broadcast packet sent by it has got corrupted and it needs to rebroadcast the same. Since, rebroadcasts can lead to redundancy and contention in the network, there must be a limit for the number of times a node can rebroadcast a packet. As [18] points out, the more times a host has heard the same broadcast packet, the less additional coverage the host will provide if it rebroadcasts the packet. The authors in [18] show that rebroadcasting a packet 3 or 4 times achieves reachability better or comparable to flooding. In our experiments we have observed similar results.

The method proposed in this paper is tested for four rebroadcast attempts. Multiple nodes that find the received broadcast to be corrupted can start transmitting CDP simultaneously. In that case, CDP from multiple nodes may overlap. However, if the transmitter of a broadcast packet detects a pulse of a minimum duration of 8µs, it identifies the pulse to be CDP and rebroadcasts the broadcast packet. The proposed method is tested over four different scenarios. Each node transmits a broadcast packet to inform about its presence to other nodes. The receivers of the broadcast packet populate their neighbor tables, which will be further used to find multiple paths from a source to a destination. Figure 1 shows the flowchart for the proposed method.

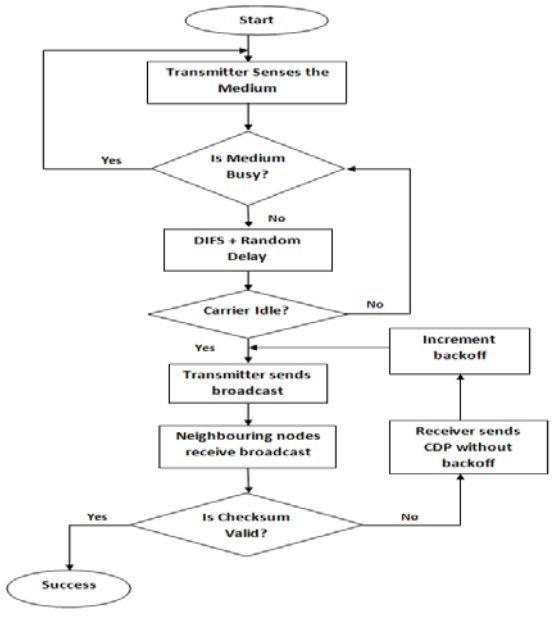

Figure 1. Flowchart for broadcast collision detection and handling

The scenarios considered to test the feasibility of the proposed scheme are as shown in Figure 2. These scenarios are simulated and the associated results are explained in detail. All these scenarios assume static topology and symmetric links. The

presence of a two-way arrow represents a symmetric link between two nodes while its absence represents the absence of link.

Scenario 1: In this scenario nodes are sparsely placed as shown in Figure 2(a). Nodes A and B are in communication range of each other, while Node C is range of D and E. Nodes D and E are not in range, and therefore do not have any link between them. It is observed through the simulations that if Nodes A and C transmit a broadcast packet simultaneously, there are no collisions at any of the receivers. However, if Nodes D and E simultaneously transmit the broadcast, then the broadcast packets may collide at Node C due to hidden node problem. After identifying a corrupted packet, Node C immediately sends a Collision Detection Pulse (CDP) without any backoff. Upon detecting a CDP on the channel just after sending the broadcast, Nodes D and E become aware that the broadcast packets sent by them have got corrupted and need to be retransmitted.

Scenario 2: This scenario is shown in Figure 2(b). In this case, Nodes A and C are made to transmit broadcast simultaneously. Since Nodes A and C are placed such that they are not in range of each other, the broadcast packets transmitted by these nodes collide at Node B due to hidden node problem, while Nodes E and D receive the broadcasts without any error. After identifying a corrupted packet, Node B immediately sends a CDP without any backoff. Since Nodes E and D are not in range of Node B, this signal is received only by Nodes A and C. Upon detecting a CDP on the channel just after sending the broadcast, Nodes A and C become aware that the broadcast packets sent by them got corrupted and need to be retransmitted.

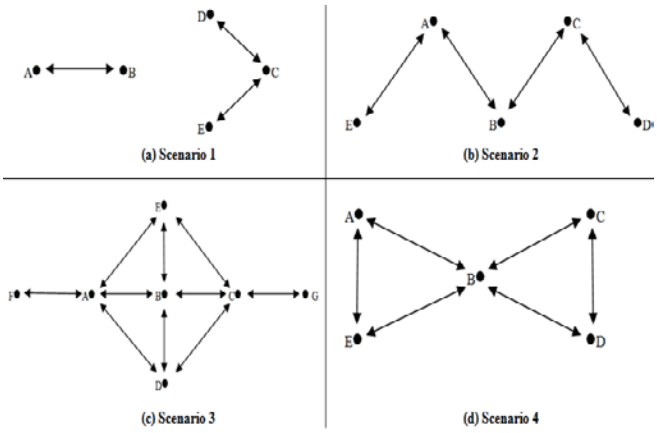

Figure 2. Scenarios to test proposed scheme

Scenario 3: This scenario is shown in Figure 2(c). Here, Nodes E, B and D are placed in range of each other and that of Nodes A and C. Node F is in range of Node A, Node G is in range of Node C, while Nodes A and C are hidden from each other. When Nodes A and C transmit a broadcast packet simultaneously, it may collide at Nodes E, B and D due to hidden node problem, while this is not the case with Nodes G and F. After identifying a corrupted packet, Nodes E, B and D send CDP without any backoff. It is observed that due to processing and propagation delays, CDP sent by the nodes tend to get delayed and overlap. However, whenever a node detects a minimum of 5µs CDP in the medium, it gets to know that previous broadcast has got corrupted. The CDP sent in this case does not affect Nodes F and G in any way. After detecting CDP in the medium, Nodes A and C become aware that the broadcast packets sent by them got corrupted and need to be retransmitted.

Scenario 4: This scenario is as shown in Figure 2(d). In this scenario Nodes A, E and B are placed such that they can communicate with each other, Nodes C, D and E are placed such that they can also communicate with each other. Specifically, Node E is placed nearer to Node A when compared to Node B, and Node D is placed nearer to Node C when compared to Node B. When Nodes A and C transmit broadcast packet, a collision is observed at B, while Nodes E and D receive the packet correctly by Nodes A and C respectively. Immediately after detecting the collision, Node B transmits CDP. Apart from Node A and C, CDP is also detected by Nodes E and C. There are two important possibilities in this case: (a) Either node E or D or both the nodes already completed receiving broadcast packet, with valid checksum and processed the packet. In such a case when any or both the nodes detect CDP in the medium they ignore it. (b) Either node E or D or both the nodes are still receiving the broadcast while Node B sends CDP. In this case the nodes receive a corrupted broadcast packet due to collision between CDP and broadcast packet in the medium. After the reception of corrupted broadcast packet, Node E or D or both nodes E and D transmit a CDP. CDP from all the nodes may overlap. Nodes A and C detect a CDP of minimum 5µs and realize that they need to retransmit the broadcast. It must be noted that in this case the CDP affects the network operation negatively. This can be perceived as the disadvantage of the proposed technique. The process of retransmission for all the scenarios is according to the flowchart shown in Figure 1.

4. SIMULATION AND ANALYSIS

The proposed method is simulated and analyzed using simulator developed in C++. All the nodes in the network broadcast 'HELLO' packets to notify their presence in the network. These broadcasts are used by the receivers for neighbor table generation. If the broadcast packets collide, neighbor tables will be incomplete, thus affecting the routing of information. Nodes which could not be populated in the neighbor table due to collision of broadcast packets are termed as 'missed nodes'.

Before transmission of the broadcast packet, the transmitter waits for 50µs and a random interval of time. A random value is generated between 31 and 1023, since *CWmin* is 31and *CWmax* is 1023. Since each backoff time slot is considered to be 20µs long, the generated random number is multiplied with 20µs to calculate the random backoff time. With every retry, the transmitter increases its backoff in steps of 10µs. Therefore, for the very first attempt, the waiting period for the transmitter is as shown in Equation (1).

$$50\mu s + 20\mu s \times (\text{rand}(31, 1023)) \quad (1)$$

After every retry, the transmitter increases the backoff period by 10 µs, as shown in Equation (2).

$$50\mu s + (n \times 10\mu s) + 20\mu s \times (\text{rand}(31, 1023)) \quad (2)$$

where 'n' varies from 1 to 4

The proposed scheme is tested and verified over grid topology since it is deterministic and covers all the four scenarios discussed in Section 3. Being deterministic, Grid topology is also easy to analyze. The scenario specifications are given in Table 1. Keeping the area of terrain same, while increasing the node density, increases the possibility of collisions. This was considered to be ideal scenario to study the performance of the proposed scheme.

Table 1. Scenario Specifications

Parameter	Value
Channel Bit Rate	2Mbps
Transmission Range	100m
Broadcast Frame Length	208 bits
DIFS Duration	50µs
Single Slot Time	20µs
Minimum Contention Window (***CWmin***)	31 Slots
Maximum Contention Window (***CWmax***)	1023 Slots
Incremental Backoff	10µs
CDP Duration	8µs
Terrain Size	300m X 300m
Node Placement	Grid
Grid Size	n x n (n = 4 to 16)

DCF is the fundamental MAC technique for IEEE 802.11 based wireless networks. It is observed in [5] that hidden terminal have a detrimental effect on the performance of IEEE 802.11 based wireless networks. The authors of [5] have observed that the performance of the protocol drops sharply with an increase in the number of hidden terminals in the network.

The performance of the proposed method is compared with that of IEEE 802.11 DCF. The scenario with single transmission attempt (Attempt 1) is equivalent to IEEE 802.11 DCF, since it does not have a provision for sending a CDP on collision detection. In other attempts, nodes identify the collision among broadcast packets sent by them through CDP and retransmit the broadcast packets.

Based on the specifications of the given scenario, results are plotted for number of 'missed nodes' (due to collision of broadcast packets) while forming the neighbor table, with varying node density. It is observed that as the node density increases, the number of nodes missed in the neighbor table due to broadcast collisions also increase. The simulations were conducted five times for each attempt with different seed values and an average of obtained results was taken.

Results obtained through simulation are shown in Figure 3. From the obtained results it is observed that the number of neighbors not discovered by a node due to broadcast collisions (number of 'missed nodes') reduces drastically at second attempt itself. Also, by the third attempt a node would have discovered most of the neighbors. This leads to a well populated neighbor table thus enhancing a node's capability to find multiple paths to a destination.

The delay introduced by the proposed scheme was also calculated, and it was observed that for all the considered grid sizes (4 x 4 to 16 x 16), all the neighbors of every node were discovered within a maximum delay of 161ms.

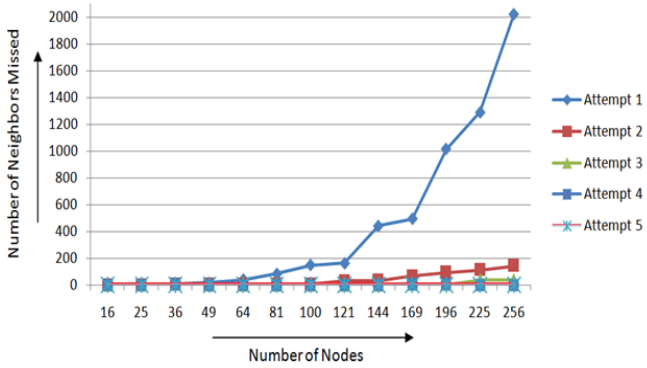

Figure 3. Variation in number of neighbors missed with varying node density

It was also observed that by 3rd attempt, i.e., at the end of 2nd broadcast, all the neighbors for every node were discovered.

5. CONCLUSION

This paper proposes a method to handle the collision of broadcast packets due to hidden terminal problem, for efficient neighbor discovery. Collision Detection Pulse (CDP) is used to make the transmitting nodes aware of a collision of broadcast packet transmitted by them. The nodes are placed in grid topology and the proposed method is tested through simulation. Results are plotted for number of 'missed nodes' (due to collision of broadcast packets) while forming the neighbor table, with varying node density. It is observed that as the node density increases, the number of nodes missed in the neighbor table due to broadcast collisions also increase. With the proposed method, every node can discover all its neighbors within a maximum delay of 161ms. Efficient neighbor discovery helps in establishing multiple paths from a source to destination.

6. REFERENCES

[1] Alagar,S.,Venkatesan,S. and Cleaveland,J.R. Reliable boradcast in mobile wireless networks. *Military Communication Conference MILCOM'95*, (San Diego, CA, 1995), Conference Record, IEEE, 236-240.

[2] Bhargavan,V.,Demers,A.,Shenker,S., and Zhang, L. MACAW: A media access protocol for wireless LANs, *Proceedings of ACM SIGCOMM-8/94*, (London England UK, 1994). 212-225.

[3] Fullmer, C.J. and Garcia, J.J. Floor acquisition multiple access (FAMA) for packet-radio networks. *Proceedings of ACM SIGCOMM, Computer Communication Review,* 1995, 262-273.

[4] Haas,Z.J. and Deng,J.Dual Busy Tone Multiple Access (DBTMA): A Multiple Access Scheme for Ad-hoc Networks. *IEEE Transactions on Communications, 50(6),* June 2002, 975-985.

[5] Khurana, S., Kahol, A. and Jayasumana, A.P. Effect of Hidden Terminals on the Performance of IEEE 802.11 MAC Protocol. *In Proceedings of 23rd Annual Conference on Local Computer Networks (LCN),* 1998, 12-20.

[6] Lim, H. and Kim,C. Multicast tree construction and flooding in wireless ad-hoc networks. In MSWIM'00: Proceedings of the 3rd ACM international workshop on modeling, analysis and simulation of wireless and mobile systems, (NewYork, USA, 2000), ACM Press, 61-68.

[7] Macker, J. Simplified multicast forwarding for manet. IETF Internet-Draft, March 2007.

[8] Pagani,E. and Rossi,G.P. Reliable broadcast in mobile multihop packet networks. *Proceedings of The Third Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'97),* (Budapest Hungary 1997), 34-42.

[9] Pagani,E. and Rossi,G.P. Providing reliable and fault tolerant broadcast delivery in mobile ad-hoc networks. *Journal Mobile Networks and Applications, 4(3),* (Hingham, MA, USA, 99), Kluwer Academic Publishers, 175-192.

[10] Park S., Yoo S.-M. An efficient reliable one-hop broadcast in mobile ad-hoc networks, 2012, Ad Hoc Netw. http://dx.doi.org/10.1016/j.adhoc.2012.03.021

[11] Peng, W. and Liu, X,-C. On the reduction of broadcast redundancy in mobile ad hoc networks. *Proceedings of the 1st ACM international symposium on Mobile ad hoc networking and computing, MobiHoc'00, IEEE Press, (Piscataway, NJ, USA, 2000)*, 129-130

[12] Peng, W. and Liu, X. AHBP: An efficient broadcast protocol for mobile ad hoc networks. *Journal of Computer Science and Technology, Volume 16, Number 2*, 2001, 114-125.

[13] Qayyum, A., Viennot, L., and Laouiti, A. Multipoint relaying: An efficient technique for flooding in mobile wireless networks. Research report RR-3898, INRIA, 2000, 1-19.

[14] Sardouk, A., Senouci, S.M., Achir, N. and Boussetta, K, Assessment of MANET broadcast schemes in the application context of multiplayer video games. *Proceedings of the 6th ACM SIGCOMM workshop on Network and system support for games, NetGames'07,* (New York, USA, 2007), 55-60

[15] Shih, K.P., Chang, C.Y., Chen, H.C. and Chang, C.W. On Avoiding RTS Collisions for IEEE 802.11 based Wireless Ad Hoc Networks. *Proceeding of 20th International Conference on Advanced Information Networking and Applications-Volume 1, (AINA'06),* (Vienna Austria 2006), 747-752

[16] Talucci,F., Gerla, M. and Fratta L. MACA-BI (MACA By Invitation) A receiver Oriented Access Protocol for Wireless Multihop Networks. *Proceedings of IEEE Personal, Indoor, and Mobile Radio Communication (PIMRC'97)*, (Helsinki 1997), 435-439.

[17] Tobagu,F.A. and Klienrock,L. Packet switching in radio channels: Part II - hidden terminal problem in carrier sense multiple-access and busy-tone solution. *In IEEE Transactions on Communication.* 1975, 1417-1433.

[18] Tseng, Y.-C., Ni, S.-Y, Chen,Y.-S. and Sheu, J.-P. The broadcast storm problem in a mobile ad hoc network. Wireless Networks, Volume 8, Kluwer Academic Publishers, 2002, 153-167.

[19] Tseng, Y.C., Ni, S.Y. and Shih, E.Y. Adaptive approaches to relieving broadcast storms in a wireless multi-hop mobile ad-hoc network. *IEEE Transactions on Computers, 52(5),* 2003, 545-557.

[20] Williams,B. and Camp,T. Comparison of broadcasting techniques for mobile ad hoc networks. *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (Mobihoc'02), (New York USA 2002)*, 194-205.

[21] Wu,C. and Li,V.O.K Receiver-initiated busy-tone multiple access in packet radio networks. *Proceedings of the ACM Workshopon Frontiers in Computer Communication Technology (SIGCOMM'87), (New York USA 1987),* 11-13.