JOURNAL OF COMPUTING, VOLUME 2, ISSUE 8, AUGUST 2010, ISSN 2151-9617
HTTPS://SITES.GOOGLE.COM/SITE/JOURNALOFCOMPUTING/
WWW.JOURNALOFCOMPUTING.ORG 

# Efficient Contention Window Control with Two-Element Array

Ali Balador, Ali Movaghar, and Sam Jabbehdari

**Abstract**—in wireless networks, the sharing channel has limited communication bandwidth. So designing efficient Medium Access Control (MAC) protocol with high performances is a major focus in distributed contention-based MAC protocol research. IEEE 802.11 MAC protocol is the most famous standard in this area. But, this standard has a problem with adopting its backoff range based on channel status. It causes some problems in throughput and fairness in a real situation. In this paper, we propose a simple algorithm that maximizes the throughput and fairness among competing nodes. We have divided nodes into four section of our backoff range. Numerical results show improvement in all performances except end to end delay.

**Index Terms**—Backoff Algorithm, Contention Window, IEEE 802.11 DCF, Mobile Ad-Hoc Network, MAC layer

## 1 Introduction

THe rapid development in wireless communication and the growth of mobile communication and computing devices like cell phones, PDAs or laptops cause Mobile Ad-Hoc Networks (MANETs) as an important part of future ubiquitous communication. MANETs are the important part of mobile communications because they are infrastructure-less, low cost and quick deployed. This technology is useful for ubiquitous environment in offices, hospitals, campuses, airports, factories and battlefield communications.

In wireless networks, the channel is shared among network's nodes. A node should compete with other nodes that have packet in their buffers to transmit. Because of that, Medium Access Control (MAC) plays an important role in controlling channel access among nodes. Study group 802.11 was formed and introduced IEEE 802.11 standard. This standard provides detailed MAC and Physical layer (PHY) specification for wireless LANs. The MAC incorporates two different medium access methods: the compulsory Distributed Coordination Function (DCF) and the mandatory Point Coordination Function (PCF). 802.11 can operate both in DCF mode and PCF mode. Every 802.11 node should implement DCF mode, which is contention-based method and supports asynchronous data transfer on a best effort basis. On the other hand, the implementation of PCF is not mandatory in IEEE 802.11. [1]

The DCF in IEEE 802.11 is based on a Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) technique, which uses a combination of the CSMA and MACA schemes. In CSMA-based schemes, sender first senses the channel to check whether it is idle or busy. If they found channel idle, transmit their own packets. On the other hand, other nodes that could not transmit their packets should defer their own transmissions when the medium is busy. IEEE 802.11 DCF uses this routine to prevent a collision with other competing nodes.Although, collision occurs at receiving nodes. There are two known problems in wireless networks: hidden node problem and exposed node problem. [2]

As illustrated in Fig. 1, node C wants to transmit to node D but mistakenly thinks that this will interfere with B's transmission to A, so C refrains from transmitting. This problem is referred to "exposed node problem" leads to loss in efficiency.

A more serious problem is known as the hidden node problem. Assume that node A is sending data to node B. a terminal C is "hidden" when it is far away from the data source A but it is close to the destination B. without the ability to detect the ongoing data transmission, C will cause a collision at B if C starts transmitting a packet. Fig. 2 shows this problem. MAC schemes are designed to overcome these problems.

For preventing these problems IEEE 802.11 DCF uses a method to reserve the medium that other neighboring nodes can distinguish the transmission. IEEE 802.11 uses

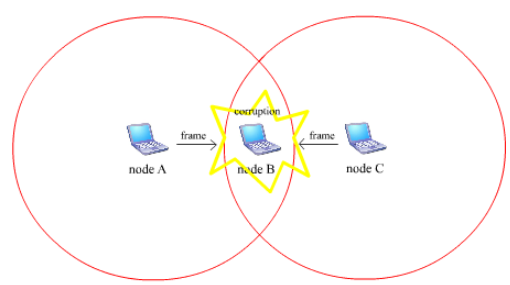


Fig. 1. Hidden node problem

Request-To-Send (RTS) / Clear-To-Send (CTS) mechan-

- *Ali Balador is with the Computer Engineering Department, Science and Research Branch, Islamic Azad University, Tehran, Iran.*
- *Ali Movaghar is with the Department of Electrical and Computer Engineering, Sharif University of Technology, Tehran, Iran.*
- *Sam Jabbehdari is with the Computer Engineering Department, North Tehran Branch, Islamic Azad University, Tehran, Iran.*

ism for this reason. The IEEE 802.11 protocol uses the RTS-CTS-DATA-ACK sequence for its communication.


Fig. 2. Exposed node problem

In addition to physical carrier sensing, it also use other carrier sensing to notify neighbors about data transmission. It is called virtual carrier sensing. This implemented in the form of a Network Allocation Vector (NAV) which is maintained by every node. The NAV indicates the amount of time that must elapse until the current transmission is complete and the medium can be checked again for idle status.

The IEEE 802.11 DCF controls priority access to the channel with different time intervals between the transmissions of frames that known as an Inter Frame Space (IFS) time intervals.

This standard determines three kinds of these intervals. It names shortest one as a Short IFS (SIFS) that has the highest priority access to the channel and is used for control packets. Secondly, DCF-IFS (DIFS) is another time interval that is used in the basic access method in IEEE 802.11 DCF. The other time interval is PCF-IFS (PIFS) that IEEE 802.11 uses it for polling in PCF mode. Finally, Extended Inter Frame Space (EIFS) interval has the lowest priority access. After an erroneous frame is detected, a node must remain idle for at least an EIFS interval. Fig. 3 shows different IFS time intervals. [3]

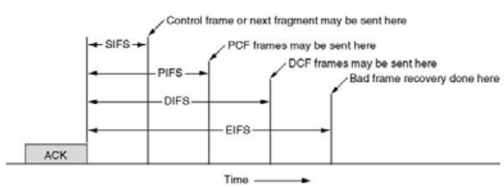


Fig. 3. Inter Frame Space time intervals.

According to DCF, before a node initiates a transmission, it senses the channel to found that another node is transmitting or not. If the medium is sensed to be idle for longer than DIFS, the node continues with its transmission. A node should use Binary Exponential Backoff (BEB) algorithm. The BEB algorithm uniformly selects the backoff time in the interval (0, CW). First of all, DCF sets CW with predefined value CWmin that is very affected on total throughput.

In other times, it is doubled with transmission failure up to another predefined value CWmax. When a node distinguishes several failures and reaches to CWmax, it keeps its value. If a node catches a channel before sending data packet, it should send RTS to notify receiver about data transmission and by receiving a CTS packet, it can start data packet transmission. Transmitter must wait SIFS time interval after receiving the CTS packet then begins to send its own data packets.

IEEE 802.11 DCF rapidly decreases the CW value to CWmin with sudden successful transmission. Unfortunately, it reduces the performance. When the backoff timer decreased by one the node checks the channel. When the channel is sensed busy, the backoff controller pauses the timer and it is resumed when the channel sensed idle again for more than DIFS. Fig. 4 shows 802.11 DCF completely.

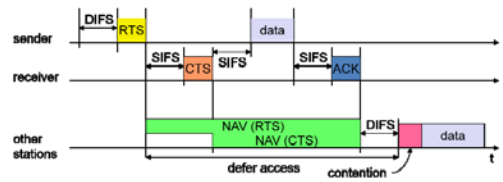


Fig. 4. IEEE 802.11 DCF mechanism

Each node has one antenna and when it sends data packet, it cannot listen to the medium. After sending data packet, transmitter should wait and listen to the channel to receive a Positive Acknowledgement (ACK) from the receiver. Receiving successful ACK packet means the receiver received the data packet successfully. After receiving correct data packet by receiver, the receiver waits for SIFS time then transmit an ACK packet to the transmitter.

If the transmitter receives the ACK packet correctly then the CW value set to its initiate value and drops the data packet. Fig. 4 shows the IEEE 802.11 DCF mechanism. Otherwise sender distinguish a collision, it increases the CW value for next competition or drops the packet if its attempts to retransmission reach predefined value. (Transmission attempt limit)

## 2 RELATED WORKS

The BEB algorithm is widely used in MAC layer protocols because of its simplicity. In this scheme, each node in order to a collision doubles its CW value up to the CWmax and sets CW value to its initial after a successful transmission:

$CW \leftarrow \min(2*CW, CWmax)$ upon collision (1)

$CW \leftarrow CWmin$ upon success (2)

As we have pointed out, the BEB scheme has problem when network size is large because it suddenly sets the CW value to CWmin upon a successful transmission.

To the best of our knowledge, numerous papers have been conducted on improving the performance of IEEE 802.11 DCF by modifying the BEB algorithm. Because of a vital problem in decreasing the CW value based on the successful transmission, many of researches in this field focus on introducing new mechanism instead of original one in BEB algorithm. Based on our research we can categorize these methods into four groups.

First group chooses the static scale for decreasing the CW value upon a successful transmission. Important factor of this group is the lowest overhead. These methods can be implemented with a bit modification in IEEE 802.11. But they do not pay attention to the network load.

This group involved methods such as MILD [4], DIDD [5]-[7], EIED [8, 9] and [10, 11].

The Multiplicative Increase the Linear Decrease (MILD) algorithm was introduced to eliminate this problem in the BEB scheme. MILD scheme increases the CW value by multiplying by 1.5 and decreases the CW by one unit.

CW ← min (1.5*CW, CWmax) upon collision (3)

CW ← CWpacket upon overhearing successful packets (4)

CW ← max (CW-1, CWmin) upon success (5)

This algorithm is conservative, as a result, it has low throughput when network load is small. But it has better performances than BEB algorithm when the network load becomes large.

DIDD is other method in this group that has better performances in high network load the same as BEB because it reduces the CW value multiplicatively upon a success.

CW ← min (CW*2, CWmax) upon a collision (6)

CW ← max (CW/2, CWmin) upon a success (7)

Group two introduced analytical models to evaluate the performance of IEEE 802.11. Methods of this group proposed mechanisms to tune the contention window size based on the estimated number of nodes by observing the channel status. It is clear that these methods bring high overhead because of their complex computations which are not acceptable in mobile communications.

For example, [12]-[18] are involved in this group. In such cases, estimation of the number of nodes (or active nodes) is the main point for adapting CW value with network load.

Up to now in most of cases, methods only change the upper bound of CW range but methods that are gathered in third group modify both upper and lower bounds of the backoff range. In addition, in these algorithms backoff range is divided into several small ranges. [19, 20] are involved in this group.

For example in DCWA [19] the CW range divided into ranges that the length of sub ranges is based on network status.

CWlb (n) = CWub (n-1) (8)

CWub (n) = CWub (n-1) + size (9)

Size = 32*n (10)

N is contention stage.

Last group contains algorithms that have planned to modify the CW value for any node that overhearing a collision such as FCR [21] and LMILD [22].

## 3 PROPOSED ALGORITHM

The IEEE 802.11 DCF changes the CW value to obtain a better value in order to decrease the collision possibility. To reach this goal, IEEE 802.11 increases the CW value upon a collision. Because collision occurance means that the network load is high and it should wait much more time. When 802.11 increases the backoff range, the collision possibility comes down and it causes better performances.

But, the main problem of IEEE 802.11 is the way that used for decreasing the CW value after a successful transmission. After each successful transmission, the CW value will reset to the initial value (CWmin) regardless the history of network condition or number of active nodes. When the number of contending nodes increases the network performances decreases significantly because of sudden reduction of contention window. As a result, we need a smooth decrease and adapted to network condition.

Fairness problem is another IEEE 802.11 problem. Because after a successful transmission it reset the CW value to CWmin but other nodes increases their CW values. In next competition the node that had a successful transmission has more opportunity to access to the channel again.

To tackle these issues, we propose a new mechanism. In our new mechanism we used a two-element array for saving network condition and we changed the upper and lower bounds of the backoff range in contrast to IEEE 802.11 DCF that only changes upper bound.

In another, we increase the backoff range when a node had two successful transmissions. With this action, we prepare the channel for other nodes to access to the channel.

### 3.1 Channel State Vector

In this study, we check the channel condition regularly and store the result to a vector. This vector was called CS (Channel Status). The CS vector plays an important role in our method because this vector shows the network condition.

We used is_idle ( ) function that is the original function in IEEE 802.11 DCF. If is_idle ( ) function becomes zero it means that the channel is busy and when it becomes one it means that the channel is free. When is_idle ( ) function produces a new value, we shifted array values. It causes the oldest one in the CS array is removed and the new one is stored in the array.

Length selection of the CS array is a challenging decision because if we choose longer array it has more overhead and if we choose smaller one it cannot show the network condition obviously. Based on simulation experiences, we found that two-element array can work well.

### 3.1 Changing the Backoff Range

In our scheme CWlb and CWub are lower and upper bounds of backoff range and we choose 0 and CWmin for these variables at starting status. Also, CS array is initialized with 11.

After each transmission trial, the backoff range is updated following table 1, respectively. Backoff ranges are dedicated and it causes reduction in collision possibility.

The node checks the is_idle ( ) function when it has new packet for transmission and stores it in a CS array. But the backoff range changes based on previous status that has stored in array. It leads to better performance because previous statuses are more reliable than current status.

TABLE 1
CW MODIFICATION

| Status | CW Range |
|---|---|
| 00 | CWub = 0<br>CWlb = 16 |
| 10 | CWub= 16<br>CWlb= 64 |
| 01 | CWub = 64<br>CWlb = 256 |
| 11 | CWub = 256<br>CWlb = 1024 |

Last aspect of this algorithm is that we increase the backoff range when a node had two successful transmissions in the past. It causes better fairness because other competing nodes found opportunity to access the medium.

## 4 SIMULATION RESULTS

### 4.1 Simulation Model

In this section, we study the performance of our new algorithm in comparison with IEEE 802.11 DCF by using NS-2 (version 2.28). [23]

Our simulation are based on a 1000 by 1000 meter flat space and 50 wireless nodes. Simulation time was set to 600 seconds. The size of data payload is 512 bytes and each node generates data packet at the rate of 4 packets per second. The propagation range for each node is 250 meters and channel capacity is 2 Mb/s.

We utilize random waypoint model as the mobility model. The minimum speed for the simulation is 0 m/s while the maximum speed is 20 m/s. pause time is selected 50 seconds.

### 4.2 Evaluating Metrics

Packet Delivery Ratio: the Packet Delivery Ratio (PDR) which represents the ratio between the number of packets originated by the application layer source and the packets received by the final destination.
Average End to End Delay: the average end to end delay which calculates the average time required to receive the packet.
Average Throughput: the average throughput which is the amount of data successfully received in a given time period that it is measured in kilo bits per second (Kbps).
Fairness Index: we use Jain's fairness index to evaluate the fairness among the flows. For a given set of flows of throughput ($b_1$, b2, b3,……, bn), the fairness index is defined in equation(x).

$$(\sum bi)^2/n^*(\sum bi^2) \qquad (11)$$

Where n stands for number of nodes which are participating in sending the data packets in the network and $b_i$ for the throughput of the $i^{th}$ node. Fairness index is always between 0 and 1. A lower value implies poorer farness. If the throughputs achieved by all the senders are same, then the farness is 1.

### 4.3 Simulation Results

In what follow, the performances of the proposed scheme and IEEE 802.11 DCF are compared based on simulations. Fig. 5 shows the Average End to End Delay in IEEE 802.11 DCF and the new algorithm, when the number of connections is increased from 10 up to 40.

As an illustrated in Fig. 5 we can see that there is reduction about 3.49% in End to End Delay before the number of connections reach to 24. But the proposed scheme provides a better delay in high network load especially when the number of connections is 30.

Fig. 6 depicts the Network Overhead Load. As shown in the Figure, it has been greatly increased with network load increase. The network overhead load gets into saturation status when the number of connections becomes 35. It means that the IEEE 802.11 has reached the maximum of network overhead load and cannot handle more overhead. The Network Overhead Load improvement can be as much as 41.85%.

Fig. 7 shows the packet delivery ratio as the number of connections increases. As shown in the figure, the packet delivery ratio improvement can be as much as 14.77%. It happens because of dividing competing nodes in four levels and using the array to saving the history of network condition. With these modifications the collision possibility goes down.

Fig. 8 depicts the throughput. As we expected, the proposed scheme obviously outperforms the IEEE 802.11 DCF in all cases. The average throughput improvement can be as much as 14.51%.

Fig. 9 shows the fairness in our scheme and IEEE 802.11 DCF. When comparing the fairness, it can be seen that our algorithm outperforms IEEE 802.11 in most cases, except for the case that the fairness is 30. Totally, our scheme improves the fairness about 3.18%.

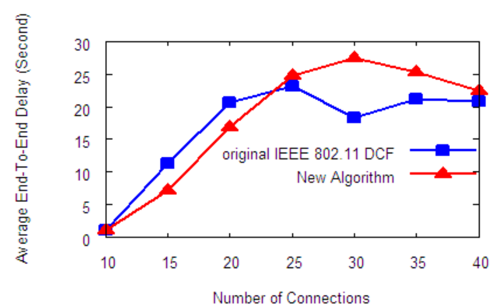


Fig. 5. Average End to End Delay.

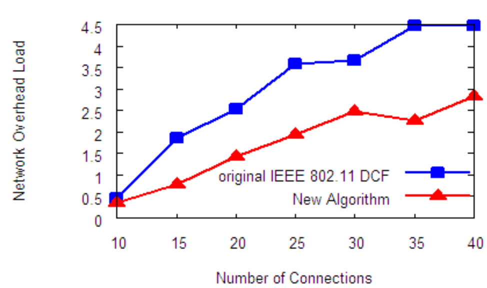


Fig. 6. Network Overhead Load

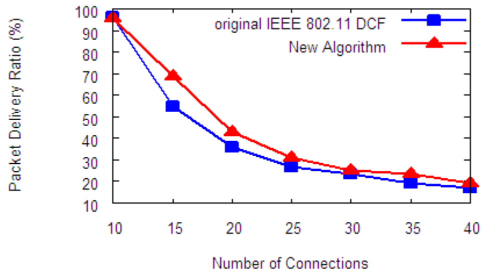


Fig. 7. Packet Delivery Ratio

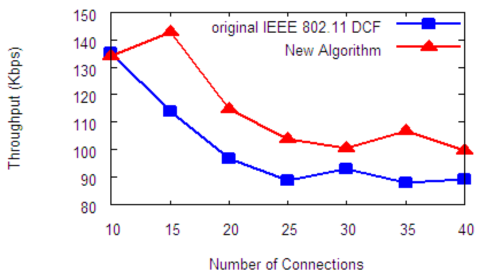


Fig. 8. Average Throughput

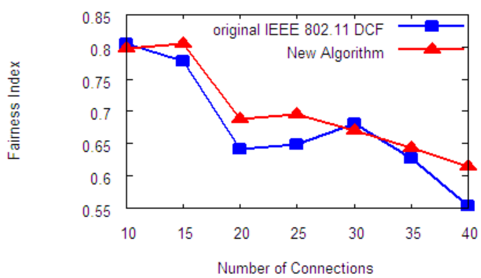


Fig. 9. Fairness Index

The fairness increases because the proposed algorithm increases the backoff range of nodes that had two successful transmissions and it makes a good opportunity for other competing nodes that cannot access to the channel in last two their attempts.

Last point is that the new algorithm has surprisingly low overhead because we used only two-element array and low computation for defining the new backoff range. It helps us so it can implement this scheme with a bit modification in 802.11.

## 5 CONCLUSION

In this paper, we have presented a simple scheme to improve IEEE 802.11 DCF performances by tuning its backoff algorithm. Our method checks the channel with a predefined array and modifies the backoff range based on array status. This is the main reason why this scheme improves performances.

In this paper we have divided nodes into four groups and allocated them specific section of our backoff range. This action has one advantage and disadvantage. In this way, we can reduce the collision possibility because only nodes that exist in the same section may collide with others. But, allocating nodes to different section causes nodes need a large amount of time to compete with others. This leads to reduction in end to end delay.

In addition, we allocate the last section to nodes that have two successful transmissions. It means that other nodes that cannot access to the channel have a good opportunity to access to the channel. As a result, we can see the fairness increase.

Extensive simulation studies for throughput, delay, NOL, packet delivery ratio and fairness have demonstrated that the proposed algorithm gives significant performances improvement compared to that for the IEEE 802.11 MAC algorithm and its simplicity is one factor that makes it suitable to implement.

## ACKNOWLEDGMENT

The authors would like to thank Dr. Ahmad Khademzadeh, who is the Head of the Iran Telecommunication Research Center, Tehran, Iran, for his comments throughout this study. We also thank Ms. Zahra Balador for providing us technical supports. The authors wish to thank the anonymous referees for their helpful comments that have significantly improved the quality of the presentation.

## REFERENCES

[1] IEEE standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, ISO/IEC 8802-11:1999(E), 1999.

[2] S. Kumar, V. S. Raghavan and J. Deng, "Medium Access Control Protocols for Ad-Hoc Wireless Networks: A Survey", *Elsevier Ad-Hoc Networks Journal*, Vol. 4,no. 3, pp. 326-358, May 2006.

[3] J. Schiller, *Mobile Communications (2nd ed.).*Addison Wesley, 2003.

[4] V. Bharghavan, A. Demers, S. Shenker and L. Zhang, "MACAW: A Media Access Protocol for Wireless LAN's", *Proc. ACM SIGCOMM Conference on Communications Architectures Protocols and Applications,* pp. 212–225, 1994.

[5] M. Natkaniec and A. R. Pach, "An analysis of the backoff mechanism used in IEEE 802.11 networks," *Proc. fifth IEEE symposium on Computers and Communications,* pp. 444–448, 2000.

[6] H. Wu, S. Cheng, Y. Peng, K. Long, and J. Ma, "IEEE 802.11 distributed coordination function (DCF): analysis and enhancement," *Proc. IEEE Int. Conference on Communications (ICC),* vol. 1, pp. 605–609, 2002.

[7] P. Chatzimisios, et al, "A simple and effective backoff scheme for the IEEE 802.11 MAC protocol ", *Proc. International Conference on Cybernetics and Information Technologies, Systems and Applications (CITSA),* vol. 1, pp. 48-53, 2005.

[8] N. Song, B. Kwak, J. Song and L.E. Miller, "Enhancement of IEEE 802.11 distributed coordination function with exponential increase exponential decrease backoff algorithm", *Proc. IEEE VTC2003 Spring,* vol. 4, pp. 2775–2778, 2003.

[9] N. Song, B. Kwak and L.E. Miller, "Analysis of EIED backoff algorithm for the IEEE 802.11 DCF", *Proc. IEEE 62nd Vehicular Technology Conference (VTC),* vol.4, pp. 2182—2186, 2005.

[10] Q. Ni, I. Aad, C. Barakat, and T. Turletti, "Modeling and Analysis of Slow CW Decrease for IEEE 802.11WLAN", *Proc. 14th IEEE International Symposium on Personal, Indoor and Mobile Ra-*

*dio Communications (PIMRC),* vol. 2, pp. 1717-1721, 2003.

[11] I. Aad, Q. Ni, C. Castelluccia, and T. Turletti, "Enhancing IEEE 802.11 performance with slow CW decrease", IEEE 802.11e working group document 802.11-02/674r0, 2002.

[12] F. Cali, M. Conti and E. Gregori, "Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit", *IEEE/ACM Trans. on Networking,* Vol. 8, No. 6, pp. 785-799, 2000.

[13] F. Calì, M. Conti and E. Gregori, "IEEE 802.11 Protocol: Design and Performance Evaluation of an Adaptive Backoff Mechanism", *IEEE Journal on Selected Areas in Communications,* Vol. 18, No. 9, pp. 1774-1786,2000.

[14] G. Bianchi, L. Fratta, and M. Oliveri, "Performance evaluation and enhancement of the CSMA/CA MAC protocol for IEEE 802.11 wireless LANs," *Proc. Seventh IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC),* vol. 2, pp. 392–396, 1996.

[15] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function", *IEEE Journal on Selected Areas in Communications,* vol. 18, no. 3, pp. 535–547, 2000.

[16] G. Bianchi, I. Tinnirello, "Kalman filter estimation of the number of competing terminals in an IEEE 802.11 Network", *Proc. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM),* vol. 2, pp. 844-852, 2003.

[17] D.J. Deng, C.H. Ke, H.H. Chen and Y.M. Huang, "Contention Window Optimization for IEEE 802.11 DCF Access Control", IEEE *Trans.* on Wireless Communications, Vol. 7, No. 12, pp. 5129-5135, 2008.

[18] L. Bononi, M. Conti, and E. Gregori, "Runtime optimization of IEEE802.11 wireless LANs performance", *IEEE Trans. on Parallel and Distributed Systems,* vol. 15, no. 1, pp. 66–80, 2004.

[19] A. Ksentini, A. Nafaa, A. Gueroui, M. Naimi, "Determinist contention window algorithm for IEEE 802.11", *Proc. IEEE 16th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC),* vol. 4, pp. 2712-2716, 2005.

[20] S. Romaszko, C. Blondia, "Dynamic distributed contention window control in wireless ad hoc LANs", *Proc. the Australian Telecommunication Networks and Applications Conference (ATNAC),* vol. 18, no. 9, 2006.

[21] J. Deng, P.K. Varshney, and Z.J. Haas, "A new backoff algorithm for the IEEE 802.11 distributed coordination function", *Proc. Communication Networks and Distributed Systems Modeling and Simulation (CNDS)*, pp. 215-225, 2004.

[22] Y. Kwon, Y. Fang, H. Latchman, "A novel MAC protocol with fast collision resolution for wireless LANs", *Proc. IEEE The 22th IEEE International Conference on Computer Communications (INFOCOM),* vol. 2, pp. 853–862, 2003.

[23] NS-2 simulator, http://www.isi.edu/nsnam/ns/ , 2009.

**Ali Balador** He was born in Iran in 8th of August 1985. He received his B.Sc. and M.Sc. degrees in computer engineering from Islamic Azad University in 2007 and 2010 respectively. His current research interests include: Wireless Networks, Ad-Hoc Networks and MAC Layer Protocol with emphasis on IEEE 802.11 standard. Balador has published a number of scientific papers in the field of Mobile Ad-Hoc Networks. He is also a Member of the Institute of Electrical and Electronics Engineers (IEEE).

**Ali Movaghar** He is currently a professor at the department of Computer Engineering in Sharif University of Technology in Tehran, Iran where he joined first as an assistant professor in 1993. He received his B.S. in Electrical Engineering from the University of Tehran in 1977 and both M.S. and Ph.D. in Computer, Information and Control Engineering from the University of Michigan at Ann Arbor in 1979 and 1985, respectively. He visited INRIA in France in 1984, worked at AT&T Bell Laboratories during 1985-1986, and taught at the University of Michigan during 1987-1989. His main areas of interest are performance and dependability modeling, verification and validation, computer networks, and distributed real-time systems. He is also a Senior Member of both the Institute of Electrical and Electronics Engineers (IEEE) and the Association for Computing Machinery (ACM).

**Sam Jabbehdari** He is currently working as an assistant professor at the department of Computer Engineering in IAU (Islamic Azad University), North Tehran Branch, in Tehran, since 1993. He received his both B.Sc. and M.S. degrees in Electrical Engineering Telecommunication from K.N.T (Khajeh Nasir Toosi) University of Technology, and IAU, South Tehran branch in Tehran, Iran, in 1988, through 1991 respectively. He was honored Ph.D. Degree in Computer Engineering from IAU, Science and Research Branch, Tehran, Iran in 2005.
He was Head of Postgraduate Computer Engineering Department IAU-North Tehran Branch during 2008-2010. He also has written "Advanced Topics in Computer Networks" book in Persian Language (Tehran, Classic, 2009). His current research interests are Scheduling, QoS, MANETs, Wireless Sensor Networks and Grid Computing Systems.