INTERNATIONAL JOURNAL OF COMMUNICATION SYSTEMS
Int. J. Commun. Syst. 2005; **18**:657–669
Published online 1 April 2005 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/dac.724

Carrier-sense-assisted adaptive learning MAC protocols for distributed wireless LANs

P. Nicopolitidis[1], G. I. Papadimitriou[1,*,†], M. S. Obaidat[2,‡] and A. S. Pomportsis[1]

[1] *Department of Informatics, Aristotle University, Box 888, 54124 Thessaloniki, Greece*
[2] *Department of Computer Science, Monmouth University, U.S.A.*

SUMMARY

A Carrier-sense-assisted adaptive learning MAC protocol for wireless LANs, capable of operating efficiently in bursty traffic wireless networks with unreliable channel feedback, is introduced. According to the proposed protocol, the mobile station that is granted permission to transmit is selected by means of learning automata. At each station, the learning automaton takes into account the network feedback information in order to update the choice probability of each mobile station. The proposed protocol utilizes carrier sensing in order to reduce the collisions that are caused by different decisions at the various mobile stations due to the unreliable channel feedback. Simulation results show satisfactory performance of the proposed protocol compared to similar MAC protocols. Copyright © 2005 John Wiley & Sons, Ltd.

KEY WORDS: wireless LANs; bursty traffic; dynamic bandwidth allocation; adaptive MAC; learning automata

1. INTRODUCTION

There are fundamental differences between wireless and wired LANs that pose difficulties in the design of medium access control (MAC) protocols for wireless LANs (WLANs) [1]. The wireless medium is characterized by bit error rates (BER) having an order of magnitude even up to ten orders of magnitude of a LAN cable's BER. Furthermore, in WLANs errors occur in bursts, whereas in traditional wired systems errors appear randomly. Finally, a fully connected topology between the nodes of a WLAN cannot be assumed. As a result, WLANs are characterized by unreliable links between nodes resulting in bursts of errors and dynamically changing network topologies.

Modern WLAN MAC protocols should be able to efficiently handle the bursty traffic that is expected to be generated by WLAN applications (such as client/server and file transfer applications between WLAN nodes). This paper proposes a carrier-sense-assisted adaptive

*Correspondence to: Georgios I. Papadimitriou, Department of Informatics, Aristotle University, Box 888, 54124 Thessaloniki, Greece.
† E-mail: gp@csd.auth.gr
‡ E-mail: obaidat@monmouth.edu

Received 30 October 2003
Revised 26 January 2005
Accepted 3 February 2005

Copyright © 2005 John Wiley & Sons, Ltd.

learning MAC protocol (CS-SAP) for Wireless LANs. The proposed protocol builds on that of AHLAP proposed in Reference [2], which utilizes learning automata in order to grant to stations permission to transmit. According to AHLAP the learning automata take into account the network feedback information in order to update the choice probability of each mobile station. It is proved [3] that the learning algorithm asymptotically tends to assign to each station a portion of the bandwidth proportional to the station's needs.

However in a wireless environment links can be highly unreliable resulting to non-common feedback for the mobile stations. Thus stations can make different decisions on which one will transmit and packet collisions will occur. To this end, CS-SAP utilizes carrier sensing in order to reduce the probability of packet collisions and increase protocol performance.

The remainder of the paper is organized as follows: Section 2 discusses work related to the subject of the paper and especially focuses on the distributed coordination function (DCF) of IEEE due to the fact that it will be used later in protocol performance comparisons. Section 3 presents the CS-SAP protocol. Simulation results comparing the performance of the proposed protocol to that of AHLAP and IEEE 802.11 are presented in Section 4. Concluding remarks are discussed in Section 5.

2. *AD HOC* MAC PROTOCOLS FOR WLANs

The first protocol to be developed for *ad hoc* wireless networks was ALOHA. The principle of ALOHA is simple: whenever each station has a data packet to transmit, it does so. Despite its simplicity, ALOHA suffers from low performance, which shows a peak of only 0.18 at an aggregate offered load of 0.5 packet/slot. Slotted ALOHA, in which data packet transmissions can occur only at distinct time points (the beginning of a slot) doubles this performance to 0.36 at an aggregate offered load of 1 packet/slot. However, it has to be noted that these peak values for both protocols do not have a practical meaning since they are accompanied by excessive network delays and protocol instability.

With the exception of ALOHA, all other distributed protocols for *ad hoc* WLANs utilize carrier sensing [4]. A problem in these protocols, stems from the fact that in general, a fully connected topology between the WLAN nodes cannot be assumed. This problem gives rise to the 'hidden' and 'exposed' terminal problems. The 'hidden' terminal problem describes the situation where a station A, not in the transmitting range of another station C, detects no carrier and initiates a transmission. If C was in the middle of a transmission, the two stations' packets would collide in all other stations B that can hear both A and C. The opposite of this problem is the 'exposed' terminal scenario. In this case, B defers transmission since it hears the carrier of A. However, the target of B, C, is out of A's range. In this case B's transmission could be successfully received by C, however this does not happen since B defers due to A's transmission. Such problems that derive from partially connected topologies have been shown to reduce the performance of carrier sense-based protocols [5, 6].

2.1. *IEEE 802.11 DCF*

The IEEE 802.11 DCF is a slotted carrier sense multiple access with collision avoidance (CSMA/CA) MAC algorithm. Thus, data transmissions can only start at the beginning of each slot. The IEEE 802.11 standard utilizes delays between successive frame transmissions, known as interframe spaces (IFS) (described later). The steps taken for channel access are as follows:

1. When a station has a packet to transmit, it first senses the medium. If the medium is sensed idle for an IFS, then the station can commence transmission immediately.
2. If the medium is initially sensed busy, or becomes busy during the IFS, the station defers transmission and continues to monitor the medium until the current transmission is over.
3. When the current transmission is over, the station waits for another IFS, while monitoring the medium. If it is still sensed idle, the station backs off a number of slots using a binary exponential backoff algorithm and again senses the medium. If it is still free, the station can commence transmission.

Of course, two or more stations can select the same slot to commence transmission, a fact that results to a collision. IEEE 802.11 DCF uses two IFS values in order to enable priority access to the channel. These are, from the shortest to the longest, the short IFS (SIFS) and the distributed coordination function IFS (DIFS). DIFS is the minimum delay for asynchronous traffic contenting for medium access. SIFS is used in for several IEEE 802.11 MAC operations such as MAC level acknowledgment (ACK), MAC frame fragmentation and the use of request-to-send/clear-to-send (RTS/CTS) control packet exchange.

The RTS/CTS mechanism enhances the two-way handshake CSMA/CA algorithm (DATA-ACK) to a four-way handshake one (RTS-CTS-DATA-ACK). When a station wants to transmit a packet, it sends a small request to send (RTS) packet to the data packet destination. The latter, if ready to receive the data packet, responds after a SIFS with a clear to send (CTS) packet allowing the sending station to commence data transmission a SIFS after the CTS reception. Stations that receive the RTS or CTS will defer until the DATA and ACK transmissions are completed.

The RTS/CTS mechanism tries to combat the 'hidden' terminal problem. The RTS and the CTS packets inform the neighbours of both communicating nodes about the length of the ongoing transmission. Stations hearing either the RTS or the CTS packet defer until the DATA and ACK transmissions are completed. RTS and CTS packets are small compared to data packets. As a result, when a collision between RTS packets occurs, less bandwidth is wasted when compared to collisions involving larger data packets. However, the use of the mechanism in a lightly loaded medium or in environments that are characterized by small data packets imposes additional delay due to the RTS/CTS overhead.

The collision avoidance part of the protocol is implemented through a random backoff procedure. As mentioned, when a station senses a busy medium, it waits for an idle SIFS period and then computes a backoff value. This value consists of a number of slots whose duration is physical layer-dependent. Initially, the station computes a backoff time ranging from 0 to 7 slots. When the medium becomes idle, the station decrements its backoff timer until it reaches zero, or the medium becomes busy again. In the latter case, the backoff timer freezes until the medium becomes idle again. When two or more station counters decrement to zero at the same time, a collision occurs. In this case, the stations compute a new backoff window given in slots by the formula $[2^{2+i} * \mathrm{ranf}()] * slot\_time$, where i is the number of times the station attempts to send the current data packet, ranf( ) a uniform variate in $(0, 1)$, $[x]$ the largest integer less than or equal to x and $slot\_time$ is the duration of a slot. Successive collisions cause the size of the backoff window, also known as contention window (CW) to increase exponentially. When it reaches a certain maximum, which is a user-defined parameter known as CWMax, i is reset to 1 and the range of the backoff window is reinitialized to 7. When a certain number of retransmissions attempts occurs for a specific frame, the frame is discarded.

3. THE CS-SAP PROTOCOL

The CS-SAP protocol utilizes learning automata [7–9]. These are structures that have been found to be useful in systems where incomplete knowledge about the environment in which those systems operate exists. In the area of data networking learning automata have been applied to a number of problems, including the design of self-adaptive MAC protocols, both for wired and wireless platforms, which efficiently operate in networks with dynamic workloads [10, 11]. Other applications of learning automata include adaptive push data broadcast systems [12] and telephone-traffic routing [13].

According to CS-SAP, each mobile station is equipped with a learning automaton which contains the normalized choice probability Π_i for each mobile station u_i in the network. The protocol operates as follows: After the network feedback is received for the transmission at slot t, at each station u_i the basic choice probabilities for slot t, $P_i(t)$, are normalized in the following way:

$$\Pi_i(t) = \frac{P_i(t)}{\sum_{k=1}^{N} P_k(t)} \tag{1}$$

$\sum_{i=1}^{N} \Pi_i(t) = 1$, where N is the number of mobile stations. Initially, the choice probabilities $P_i(0)$ are the same for all network stations, thus $\Pi_i(0) = 1/N$.

We assume a slotted channel and packets of equal lengths. We define a time slot to have the time duration necessary for a data packet transmission to successfully take place. Thus, this time interval also includes control overhead. At the beginning of each time slot, the carrier-sense mechanism is applied. This will be described in the next subsection. In what follows we focus on the way probability updating takes place in CS-SAP.

At each time slot t, the normalized probabilities $\Pi_i(t)$ are used to grant permission to transmit to a mobile station. After each slot has elapsed, the basic choice probability $P_i(t)$ of the selected station u_i is updated according to the network feedback information. If station u_i transmitted a packet during slot t, then its basic choice probability is increased. Otherwise, if station u_i was idle, its basic choice probability is decreased. The following probability updating scheme is used:

$$\begin{aligned} P_i(t+1) &= P_i(t) + L(1 - P_i(t)) \\ &\quad \text{if } u(t) = u_i \text{ and SLOT}(t) = \text{SUCCESS} \\ P_i(t+1) &= P_i(t) - L(P_i(t) - a) \\ &\quad \text{if } u(t) = u_i \text{ and SLOT}(t) = \text{IDLE} \end{aligned} \tag{2}$$

$L, a \in (0, 1)$ and $P_i(t) \in (a, 1)$. L governs the speed of the automaton convergence and the selection procedure for a value of L reflects the classic speed versus accuracy problem. The lower the value of L the more accurate the estimation made by the automaton, a fact however that comes at expense over convergence speed. The role of parameter a is to enhance the adaptivity

Copyright © 2005 John Wiley & Sons, Ltd.

of the protocol. This is because when the choice probability of a station approaches zero, then this station is not selected for a long period of time. During this period, it is probable that the station transits from idle to busy state. However, since the mobile station does not grant permission to transmit, the automaton is not capable of 'sensing' such transitions. Thus, the use of a non-zero value for parameter a prevents the choice probabilities of the stations from taking values in the neighbourhood of zero and increases the adaptivity of the protocol.

The above probability updating scheme suits well to bursty traffic conditions. This is due to the fact that when the selected mobile station had a packet to transmit, its choice probability is increased. On the other hand, if the selected station does not have buffered packets, its choice probability is reduced. This kind of behaviour suits well bursty conditions in which a packet arrival by the selected station implies back-to-back packet arrivals by the same station in the near future. It is proved [3] that the choice probability of each mobile station converges to the probability that this station is not idle.

CS-SAP updates the choice probabilities of mobile stations according to the network feedback information. It is proved [3] that the choice probability of each mobile station converges to the probability that this station is ready to transmit. Thus for any two mobile stations u_i and u_j, with d_i and d_j being their probabilities of being ready to transmit, respectively, CS-SAP asymptotically tends to satisfy the relation:

$$\frac{\Pi_i}{\Pi_j} = \frac{P_i}{P_j} = \frac{d_i}{d_j} \tag{3}$$

In order to obtain a better understanding of the claim of the above equation, we performed a simulation study for a CS-SAP WLAN of 10 mobile stations, from which only stations 1 and 2 are active, with $d_1 = 0.8$ and $d_2 = 0.5$. In order to simulate a changing environment, after some time, d_1 and d_2 change values to $d_1 = 0.5$ and $d_2 = 0.8$. The result of this experiment, which can be seen in Figure 1, shows that the automaton estimates of the basic choice probabilities P_1 and P_2, both before and after the environmental change, converge to d_1 and d_2, respectively. The same stands for the normalize choice probabilities Π_1 and Π_2 which converge to $d_1/(d_1 + d_2)$ and $d_2/(d_1 + d_2)$, respectively, in all cases. Thus, the claim of Equation (3) indeed stands.

Based on the above discussion, it is clear that in a noiseless environment, CS-SAP is collision-free. This is due to the fact that all stations use the same protocol and due to the broadcast nature of the wireless medium the network feedback is common for all stations. Therefore, at each slot, all stations choose the same station to be granted permission to transmit and the protocol is collision-free despite its distributed nature. However, in the presence of a noisy environment, it is possible that the network feedback is not common for all stations. Thus, the choice probabilities values may be different at several network nodes and thus collisions may occur. In order to avoid excessive collisions, CS-SAP piggybacks the K largest probabilities in the station's data packet and the rest of the probabilities, which obviously correspond to those stations that are not favoured to transmit at this time, take the value of a. This is done in an effort to reduce the differences between the probability distribution vectors of the various stations. The selection for the value of K depends on the number of stations N. If the network comprises a large number of stations, then a selection for K with $K < N$ will limit the overhead caused by the protocol.

Copyright © 2005 John Wiley & Sons, Ltd.

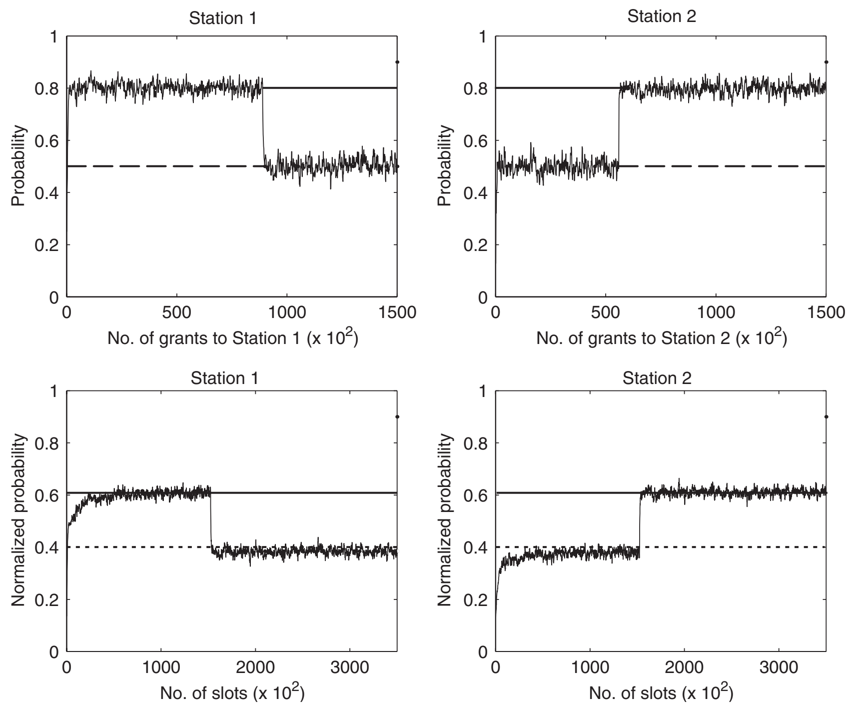

Figure 1. Convergence of basic and normalized choice probabilities for stations 1 and 2.

3.1. Collision reduction in CS-SAP

Nicopolitidis *et al.* [2] showed that the dissemination of the probability distribution vector provides satisfactory performance for the protocol compared to TDMA and IEEE 802.11 DCF. However under relatively highly unreliable wireless links a considerable number of collisions will still take place in Reference [2], despite the piggybacking mechanism. CS-SAP provides additional performance improvement by integrating a carrier-sense mechanism in order to reduce the number of collisions.

Carrier sensing in CS-SAP works as follows: At the beginning of each slot a small time window (contention window) is dedicated to contention for channel access by stations that have granted themselves permission to transmit. Recall that this is possible to be done since the automata at some stations will typically receive different network feedback for the same slot and the piggybacking mechanism does not ensure probability vector consistency at the mobile stations. The size of the contention window can be easily adjusted to consist of a number of l minislots, with $l \propto N$. Each contending station selects a random minislot r, $r \in [0..l-1]$ and transmits a burst signal until the end of the contention stage if it has not heard a burst at minislots between 0 and $r-1$. Contending stations that hear the burst back off for this time slot and thus the station that selected the lower minislot gains access to the channel.

Of course a collision can still occur in CS-SAP if (a) more than one contending stations choose the lower minislot or (b) the station that chooses the lower minislot is ‘hidden’ (out of range) of one or more other contending stations. However, as will be seen in the next section, the

Copyright © 2005 John Wiley & Sons, Ltd.

integration of the carrier-sensing scheme in networks with relatively highly unreliable links yields performance improvement over AHLAP and additional performance improvement over IEEE 802.11 DCF.

4. PERFORMANCE EVALUATION

4.1. Simulation environment

Using simulation, we compared CS-SAP against AHLAP and the IEEE 802.11 DCF. The simulator models N mobile clients and the wireless links as separate entities. Each mobile station uses a buffer to store the arriving packets. The buffer length is assumed to be equal to Q packets. Any packets arriving to find the buffer full, are dropped. Each simulation run is carried out until R packets successfully reach their destination.

The bursty traffic was modelled in the following way [11, 14, 15]: We now define 'time slot' as the time duration required for a data packet to be transmitted over the wireless link. Each source node can be in one of the two states, S_0 and S_1. When a source node is in state S_0 then it has no packet arrivals. When a source node is in state S_1 then, at each time slot, it has a packet arrival with probability Z. Given a station is in state S_0 at time slot t, the probability that this station will transit to state S_1 at the next time slot is P_{01}. The transition probability from state S_1 to state S_0 is P_{10}. It can be shown that, when the load offered to the network is R packets/slot and the mean burst length is B slots, then the transition probabilities are: $P_{01} = R/B(NZ - R)$ and $P_{10} = 1/B$.

The simulation assumes that the channel slots can be in one of the following three states: successful transmission, collision or idle. The simulator takes into account the following possibilities:

1. A 'successful' slot is perceived by a station as 'idle', due to the fact that this station is out of range of the transmitting one.
2. A 'successful' slot is perceived by a station as a 'collision' one, due to bit errors imposed by the wireless channel.
3. A 'collision' slot is perceived by a station as 'successful', due to the power capturing phenomenon.
4. A 'collision' slot is perceived by a station as 'idle', due to the fact that this station is out of range of the transmitting one.

In our simulation model, the condition of the wireless link between any two stations was modelled using a finite state machine with two states. The model, comprises the following two states:

- Stage G, denotes that the wireless link is in a relatively 'clean' condition and is characterized by a small BER, which is given by the parameter G_BER.
- Stage B, denotes that the wireless link is in a condition characterized by a high BER, which is given by the parameter B_BER.

We assume that the background noise is the same for all stations and thus the principle of reciprocity stands for the condition of any wireless link. Therefore, for any two stations A and

B, the BER of the link from A to B and the BER of the link from B to A are the same. The time periods spent by a link in states G and B are exponentially distributed, but with different average values, given by the parameters TG and TB. The status of a link probabilistically changes between the two states. When a link has spent its time in state G, the link transits to stage B. When a link has spent its time in state B, the link transits to stage G.

We employed the following broadly used performance metrics in order to compare the protocols:

1. The throughput versus offered load characteristic.
2. The delay versus throughput characteristic.

We simulated the protocols for the following two different network configurations:

1. Network N_1: $N = 10$, $Q = 10$, $B = 10$, $Z = 1.0$, $TG/TB = 1$.
2. Network N_2: $N = 10$, $Q = 10$, $B = 10$, $Z = 1.0$, $TB/TG = 3$.

In the simulations, the following parameter values remain constant: $R = 4\,000\,000$, $G\_\mathrm{BER} = 10^{-10}$, $B\_\mathrm{BER} = 10^{-4}$, $TG = 3$ s, $K = 2$, $R\_\mathrm{LIM} = 6$, $P_c = P_{iG} = 0.1$, $P_{iB} = 0.5$. R_LIM sets the maximum number of retransmission attempts per packet. P_c is the probability that when two or more data packets collide, one of them is successfully decoded at the destination station due to the power capture phenomenon. P_{iG} is the probability that a transmission from station S1 does not get through to station S2 when the link from S1 to S2 is in the G state. Similarly, P_{iB} is the probability that a transmission from station S1 does not get through to station S2 when the link from S1 to S2 is in the B state. In both cases, S2 perceives the slot as idle. The DATA packet size is set to 2000 bits and the sizes of all control packets for the protocols are set to 100 bits. The wireless medium bit rate was set to 1 Mbps and the propagation delay between any two stations was set to 0.0005 ms corresponding to inter-station distances of 150 m. The time duration of a minislot equals 0.001 ms. Finally, we did not take into account the hidden terminal problem for IEEE 802.11 DCF. Thus, stations in 802.11 are not allowed to collide with ongoing CTS-DATA-ACK packet exchanges. Despite the fact that hidden terminals are not accounted for in IEEE 802.11 DCF, the proposed protocol still performs significantly better for reasons that will be explained in the next subsection.

4.2. Simulation results

The delay versus throughput characteristics of the compared protocols when applied to networks N_1 and N_2 are shown in Figures 2 and 4, respectively, while the throughput versus offered load characteristics when applied to networks N_1 and N_2 are shown in Figures 3 and 5, respectively. The main conclusions that can be drawn from the figures are the following:

- CS-SAP provides additional performance improvement over AHLAP. This happens especially in high loads, where the increased aggregate network offered load and the unreliable feedback cause an increase in collisions.
- The reasons for AHLAP performance superiority over IEEE 802.11 DCF have also been addressed in Reference [2]. However, in an increased unreliable environment such as the one in our simulations, IEEE 802.11 is negatively dominated by the RTS-CTS-DATA-ACK four-way handshake. In order for a successfull data packet reception, all of these

Copyright © 2005 John Wiley & Sons, Ltd.

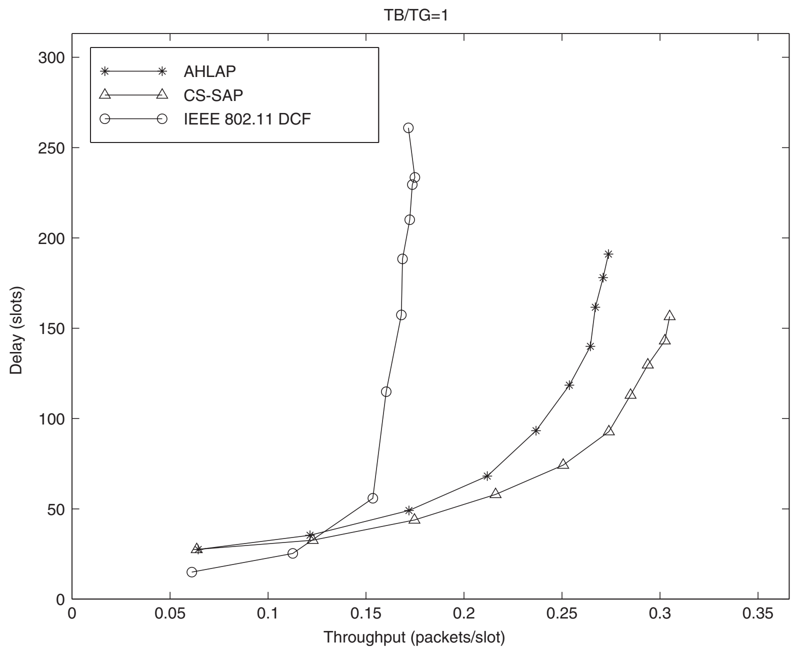

Figure 2. The delay versus throughput characteristics of CS-SAP, AHLAP and IEEE 802.11 DCF when applied to network N_1.

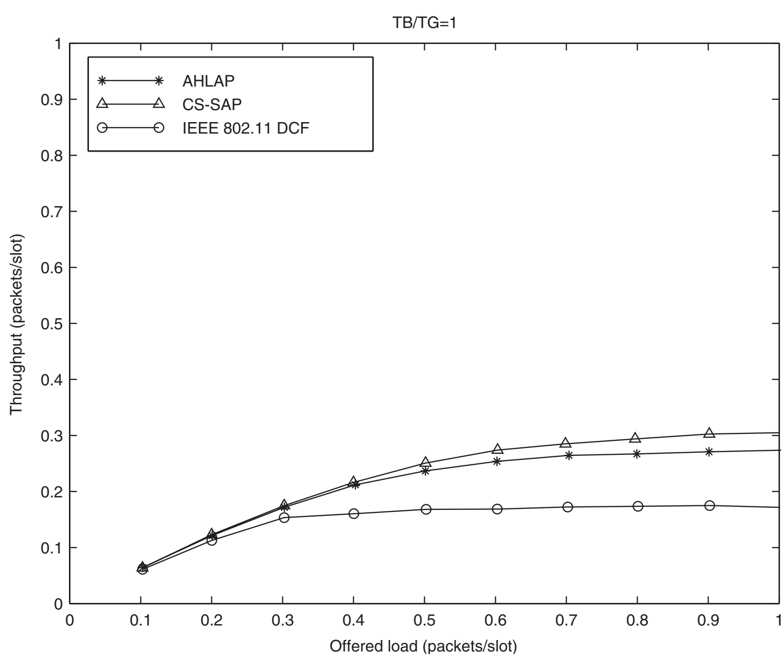

Figure 3. The throughput versus offered load characteristics of CS-SAP, AHLAP and IEEE 802.11 DCF when applied to network N_1.

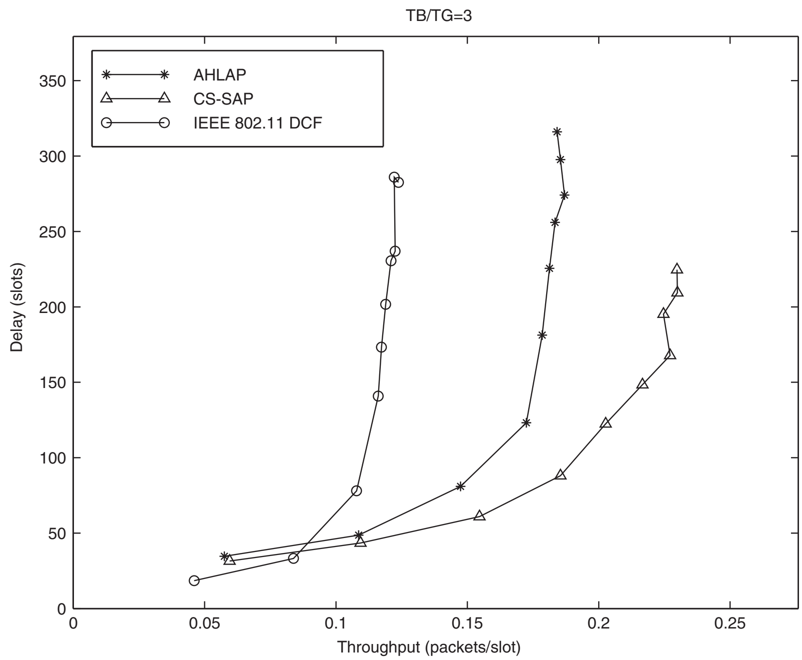

Figure 4. The delay versus throughput characteristics of CS-SAP, AHLAP and IEEE 802.11 DCF when applied to network N_2.

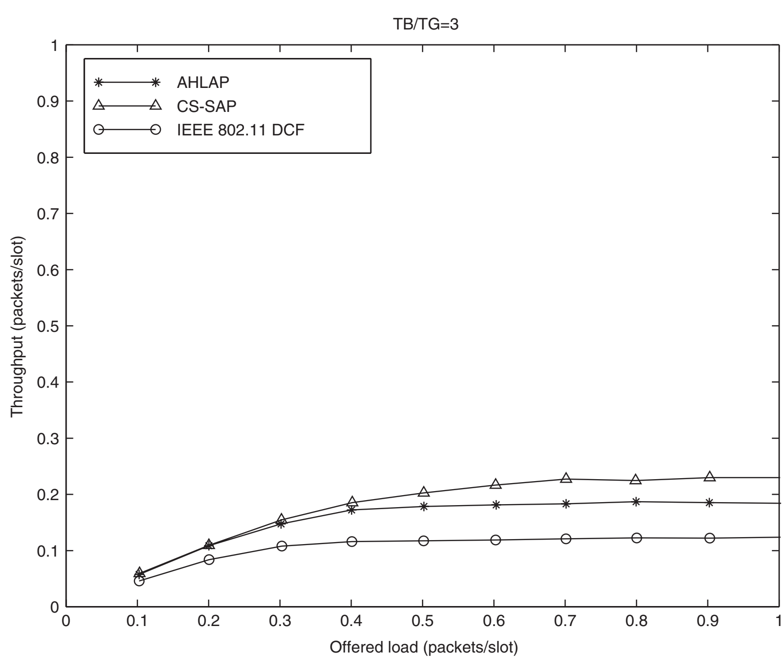

Figure 5. The throughput versus offered load characteristics of CS-SAP, AHLAP and IEEE 802.11 DCF when applied to network N_2.

Copyright © 2005 John Wiley & Sons, Ltd.

packets must be received correctly. An increased P_i (which is obviously the average of P_{iG} and P_{iB} averaged over the durations of the G and B states) affects this four-way handshake more than the two-way one of CS-SAP and AHLAP.

5. CONCLUSION

Modern WLAN MAC protocols should be able to efficiently handle the bursty traffic that is expected to be generated by WLAN applications. This paper proposes CS-SAP, an *ad hoc* learning-automata-based wireless MAC protocol utilizing carrier sense. The protocol is able to achieve significantly higher throughput and lower delay values compared to the AHLAP protocol and IEEE 802.11 DCF under bursty traffic conditions in wireless environments. Its main characteristics are:

1. It achieves a high performance, even when the offered traffic is bursty.
2. It is self-adaptive. Each station is assigned a fraction of the bandwidth proportional to its needs.
3. It is fully distributed, thus no centralized control of the network is required.
4. Due to its distributed nature, it is fault-tolerant, since its operation is not affected from a station failure.

REFERENCES

1. Nicopolitidis P, Obaidat MS, Papadimitriou GI, Pomportsis AS. *Wireless Networks*. Wiley: New York, 2003.
2. Nicopolitidis P, Papadimitriou GI, Pomportsis AS. Distributed protocols for *ad hoc* wireless LANs: a learning-automata-based approach. *Ad Hoc Networks Journal* 2004; **2**(4):419–431.
3. Papadimitriou GI, Pomportsis AS. Self-adaptive TDMA protocols: a learning-automata-based approach. *Proceedings of IEEE ICON' 99*, 1999.
4. Chandra A, Gummalla V, Limb O. Wireless medium access control protocols. *IEEE Communication Surveys* 2000; (second quarter issue).
5. Weinmiller J, Woesner H, Ebert J-P, Wolisz A. Analyzing the RTS/CTS mechanism in the DFWMAC media access protocol for wireless LAN's. *Proceedings of IFIP TC6 Workshop Personal Wireless Communications* (*Wireless Local Access*), 1995.
6. Khurana S, Kahol A, Jayasumana AP. Effect of hidden terminals on the performance of IEEE 802.11 MAC protocol. *Proceedings of LCN'98*, 1998; 12–20.
7. Najim K, Poznyak AS. *Learning Automata*: *Theory and Applications*. Pergamon: New York, NY, 1994.
8. Narendra KS, Thathachar MAL. *Learning Automata*: *An Introduction*. Prentice-Hall: Englewood Cliffs, NJ, 1989.
9. Obaidat MS, Papadimitriou GI, Pomportsis AS. (Guest eds). Learning automata: theory, paradigms and applications. *IEEE Transactions on Systems*, *Man and Cybernetics—Part B* 2002; **32**(6):(special issue).
10. Papadimitriou GI, Pomportsis AS. Learning automata-based TDMA protocols for broadcast communication systems with bursty traffic. *IEEE Communication Letters* 2000; **4**(3):107–109.
11. Nicopolitidis P, Papadimitriou GI, Pomportsis AS. Learning automata-based polling protocols for wireless LANs. *IEEE Transactions on Communications* 2003; **51**(3):453–463.
12. Nicopolitidis P, Papadimitriou GI, Pomportsis AS. Using learning automata for adaptive push-based data broadcasting in asymmetric wireless environments. *IEEE Transactions on Vehicular Technology* 2002; **51**(6): 1652–1660.
13. Narendra KS, Thathachar MAL. *Learning Automata*: *An Introduction*. Prentice-Hall: Englewood Cliffs, NJ, 1989.
14. Jain R, Routhier S. Packet trains—measurements and a new model for computer network traffic. *IEEE Journal on Selected Areas in Communications* 1986; **SAC-4**(6):986–995.
15. Danielsen SL, Joergensen C, Mikkelsen B, Stubkjaer KE. Analysis of a WDM packet switch with improved performance under bursty traffic conditions due to tuneable wavelength converters. *IEEE Journal of Lightwave Technology* 1998; **16**(5):729–735.

Copyright © 2005 John Wiley & Sons, Ltd.

AUTHORS' BIOGRAPHIES

Petros Nicopolitidis received the BS and PhD degrees in Computer Science from the Department of Informatics, Aristotle University of Thessaloniki, in 1998 and 2002, respectively. Since 2004 he is a Lecturer at the same Department. His research interests are in the areas of Wireless Networks and Mobile Communications. He is co-author of the book Wireless Networks (New York: Wiley, 2003).

Georgios Papadimitriou received the Diploma and PhD degrees in Computer Engineering from the University of Patras, Greece in 1989 and 1994, respectively. From 1989 to 1994 he was a Teaching Assistant at the Department of Computer Engineering of the University of Patras and a Research Scientist at the Computer Technology Institute, Patras, Greece. From 1994 to 1996 he was a Post-doctorate Research Associate at the Computer Technology Institute. From 1997 to 2001, he was a Lecturer at the Department of Informatics, Aristotle University of Thessaloniki, Greece. Since 2001 he is an Assistant Professor at the Department of Informatics, Aristotle University of Thessaloniki, Greece. His research interests include optical networks, wireless networks, high-speed LANs and learning automata. Prof. Papadimitriou is Associate Editor of five scholarly journals, including the IEEE Transactions on Systems, Man and Cybernetics—Part C, the IEEE Transactions on Broadcasting, and the IEEE Communications Magazine. He is co-author of the books 'Multiwavelength Optical LANs' (Wiley, 2003) and 'Wireless Networks' (Wiley, 2003) and co-editor of the book 'Applied System Simulation' (Kluwer, 2003). He is the author of more than 100 refereed journal and conference papers. He is a Senior Member of IEEE.

Professor Mohammad S. Obaidat is an internationally well known academic, researcher, and scientist. He received his PhD and MS degrees in Computer Engineering with a minor in Computer Science from The Ohio State University, Columbus, Ohio, U.S.A. Dr Obaidat is currently a full Professor of Computer Science at Monmouth University, NJ, U.S.A. Among his previous positions are Chair of the Department of Computer Science and Director of the Graduate Program at Monmouth University and a faculty member at the City University of New York. He has received extensive research funding and has published about three hundred refereed technical articles in scholarly international journals and proceedings of international conferences. He has written five books and over three hundred refereed scholarly journal and conference articles. Dr Obaidat has served as a consultant for several corporations and organizations worldwide and is editor/associate editor of many scholarly journals including the IEEE Transactions on Systems, Man and Cybernetics—Parts A and B, Computer Communications, International Journal of Communication Systems, Computers and EE, Simulation: Transactions of the Society for Modelling and Simulation International, The Journal of Supercomputing, and Journal of Defense Modelling and Simulation. In 2002, he was the scientific advisor for the World Bank/UN Workshop on Fostering Digital Inclusion. He was an IEEE (Institute of Electrical and Electronics Engineers) Distinguished Visitor/Speaker and has been serving as a National ACM (Association for Computing Machinery) distinguished Lecturer since 1995. Recently, Dr Obaidat was awarded the distinguished Nokia Research Fellowship and the Distinguished Fulbright Scholarship. Dr Obaidat has made pioneering and lasting contributions to the multi-facet fields of computer science and

Copyright © 2005 John Wiley & Sons, Ltd.

engineering. He has guest edited numerous special issues of scholarly journals such as IEEE Transactions. Obaidat has served as the steering committee chair, advisory Committee Chair and program chair of many international conferences. He is the founder of the International Symposium on Performance Evaluation of Computer and Telecommunication Systems, SPECTS and has served as the General Chair of SPECTS since its inception. Obaidat has received a recognition certificate from IEEE. Between 1994 and 1997, Obaidat has served as distinguished speaker/visitor of IEEE Computer Society. Since 1995 he has been serving as an ACM distinguished Lecturer. Between 1996 and 1999, Dr Obaidat served as an IEEE/ACM program evaluator of the Computing Sciences Accreditation Board/Commission, CSAB/CSAC. Between 1995 and 2002, he has served as a member of the board of directors of the Society for Computer Simulation International. Between 2002 and 2004, he has served as Vice President of Conferences of the Society for Modelling and Simulation International SCS. Prof. Obaidat is currently Vice President of Membership of the Society for Modelling and Simulation International SCS. He has been invited to lecture and give keynote speeches worldwide. His research interests are: wireless communications and networks, performance evaluation of computer systems, algorithms and networks, telecommunications and Networking systems, high performance and parallel computing/computers, applied neural networks and pattern recognition, information and computer security, security of e-based systems, and speech processing. He is currently on sabbatical leave as Fulbright distinguished Professor and Advisor to the President of Philadelphia University for Research, Development and Information Technology, Jordan. Prof. Obaidat is a Fellow of the Society for Modelling and Simulation International SCS, and a Fellow of the Institute of Electrical and Electronics Engineers (IEEE).

Andreas S. Pomportsis is received the BS degree in Physics and the MS degree in Electronics and Communications from the University of Thessaloniki, Greece, and the Diploma degree in Electrical Engineering from the Technical University of Thessaloniki, Greece. In 1987, he received the PhD degree in Computer Science from the University of Thessaloniki. Currently, he is a Professor at the Department of Informatics, Aristotle University of Thessaloniki, Greece. He is co-author of the books Wireless Networks (New York: Wiley, 2003) and Multiwavelength Optical LANs (New York: Wiley, 2003). His research interests include Computer Networks, Learning Automata, Computer Architecture, Parallel and Distributed Computer Systems, and Multimedia Systems.

Copyright © 2005 John Wiley & Sons, Ltd.