A Power-Efficient Approach to Adaptive Polling Protocols for Wireless LANs

P. Nicopolitidis, G. I. Papadimitriou, *Senior Member, IEEE*, T. D. Lagkas, and A. S. Pomportsis

***Abstract*— This paper proposes the Low Power Adaptive Polling (LPOAP) MAC protocol for infrastructure Wireless LANs (WLANs). It operates efficiently under bursty traffic and can reduce mobile power consumption without performance penalties.**

***Index Terms*— Low power MAC, adaptive polling, learning automata.**

I. Introduction

According to LPOAP, which is based on the LEAP protocol [1], the mobile station that will transmit is selected by the Access Point (AP) by means of a pursuit Learning Automaton. The AP uses the network feedback information to update the choice probabilities of the mobile stations. We implement the low-power option via utilizing a control packet to reduce power consumption in medium to high network loads.

II. The LPOAP Protocol

In the LPOAP protocol, the AP is equipped with a Continuous Pursuit Reward-Penalty (CP_{R-P}) Learning Automaton [2]. Pursuit schemes derive from the classical Learning Automata and have been shown to be more efficient in terms of convergence speed. According to the proposed scheme the AP contains the normalized choice probability $P_i(j)$ for each mobile station i under its coordination. According to the CP_{R-P} learning mechanism the AP also keeps a vector d that contains the running estimates of the environmental reward for the selection of each mobile station.

At the beginning of each polling cycle j, the AP polls according to the normalized choice probabilities. The protocol uses four control packets, POLL, NO_DATA, BUFF_DATA and ACK whose durations are t_{POLL}, t_{NO_DATA}, t_{BUFF_DATA} and t_{ACK} respectively.

As far as power consumption at the radio level is concerned, we consider that each station can be at one of the following power states: a) Transmit (TRM) state, b) Receive (REC) state, in which a mobile station operates when it receives a packet either with or without bit errors, c) Idle (IDLE) state, in which a mobile station operates when it neither transmits or receives but keeps its transceiver switched on and d) Doze (DOZE) state, in which a mobile station operates when it turns the electrical circuit of its transceiver off.

Initially, all mobile stations are in the IDLE state. Assuming that the AP polls mobile station k at time position t which marks the beginning of polling cycle j, the propagation delay is t_{PROP_DELAY} and a station's DATA transmission takes t_{DATA} time to complete, the following events are possible:

1) The poll is received at station k at time $t + t_{POLL} + t_{PROP\_DELAY}$. Then:
   - If station k does not have a buffered packet, it immediately responds to the AP with a NO_DATA packet. If the AP correctly receives the NO_DATA packet it immediately proceeds to poll the next station. This poll is initiated at time $t + t_{POLL} + 2 * t_{PROP\_DELAY} + t_{NO\_DATA}$. In case of no reception at the AP the next poll begins at time $t + t_{POLL} + 4 * t_{PROP\_DELAY} + t_{BUFF\_DATA} + t_{DATA} + t_{ACK}$. In both of the above cases the Automaton in the AP increases the choice probability of station v having the highest current reward estimate d_v and then lowers the value of the reward estimate d_k of station k. The environmental response $b(j)$ in this case equals 1.
   - If station k has a buffered DATA packet, it responds to the AP with a BUFF_DATA packet with the address $addr$ of the receiver of its DATA packet piggy-backed on BUFF_DATA. Every mobile station with address other than $addr$ that receive BUFF_DATA will know that there is no reason to retain its radio transceiver on, so it goes to the DOZE state and will return to the IDLE state after the time interval that is needed for the DATA packet to be successfully delivered. Station k transmits the DATA packet to its destination and waits for an acknowledgment (ACK) packet. After the poll, the AP monitors the wireless medium for a time interval equal to $t_{BUFF\_DATA} + t_{DATA} + t_{ACK} + 3 * t_{PROP\_DELAY}$. If it correctly receives one or more of BUFF_DATA, DATA, ACK, it concludes that station k received the poll and has one or more buffered data packets. Thus, the AP raises the choice probability of station v having the highest current reward estimate d_v and then raises the value of the reward estimate d_k of station k. The environmental response $b(j)$ in this case equals 0. However, if the AP does not receive feedback, it concludes that it cannot communicate with station k and proceeds with the next poll at time $t + t_{POLL} + 4 * t_{PROP\_DELAY} + t_{BUFF\_DATA} + t_{DATA} + t_{ACK}$. In that case the Automaton in the AP will increase the choice probability of station v with the highest current reward estimate d_v and then lower the value of the reward estimate d_k of station k. In this case $b(j)$=1.

Manuscript received February 4, 2007. The associate editor coordinating the review of this letter and approving it for publication was Dr. Nikos Nikolaou.

The authors are with the Department of Informatics, Aristotle University, Box 888, 54124 Thessaloniki, Greece (email: petros@csd.auth.gr).

Digital Object Identifier 10.1109/LCOMM.2007.070184.

2) The poll is not received at station k, k does not respond to the AP and the AP proceeds to poll the next station at time $t + t_{POLL} + 4 * t_{PROP\_DELAY} + t_{BUFF\_DATA} + t_{DATA} + t_{ACK}$. In this case the AP increases the choice probability of station v with the highest current reward estimate d_v and then lowers the value of the reward estimate d_k of station k. In this case again $b(j)$=1.

From the above discussion, it is obvious that the learning algorithm takes into account both the bursty nature of the traffic and the bursty appearance of errors over the wireless medium. The CP_{R-P} algorithm is employed after each poll and is described in Figure 1. N is the number of mobile stations, L is the learning speed parameter, $0 < L < 1$, $d_i(j)$ is a vector with reward estimates for each station i at cycle j, v is the index of the maximum value of $d(j)$, e_v is a unit N-vector with 1 in the m^{th} coordinate, $W_i(j)$ is the number of times the i^{th} station has been rewarded up to cycle j, $Z_i(j)$ is the number of times the i^{th} station has been selected up to cycle j, and $b(j)$ is the environmental response after polling station k at cycle j.

Initialize P_i=1/*N*, $1 \le i \le N$.
Initialize vector *d* by choosing each station a few times
Repeat
Step 1: At cycle *j* select station *k* according to the probability distribution *P(j)*.
Step 2: If *v* is the station with the highest reward estimate update *P*(*j*): *P*(*j*+*1*)=(1-*L*)**P*(j) + *L**e_v
Step 3: Update $d_k(j)$ (the reward estimate of *k*)
$W_k(j+1) = W_k(j) + (1 - b(j))$
$Z_k(j+1) = Z_k(j) + 1$
$d_k(j+1) = W_k(j+1)/Z_k(j+1)$

Fig. 1. The CP_{R-P} algorithm.

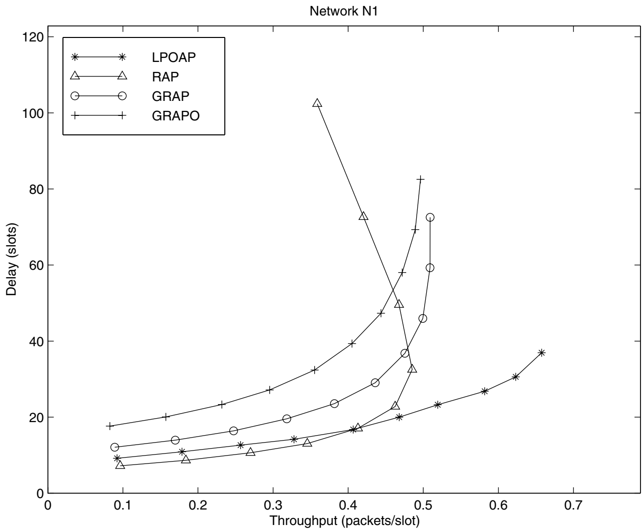

Fig. 2. The delay versus throughput characteristics of LPOAP, RAP, GRAP and GRAPO when applied to network N1.

III. Simulation Results

Using simulation, we assessed a) the performance of LPOAP against the RAP, GRAP and GRAPO ([3], [4]) polling protocols in bursty WLANs and b) the gain in power efficiency in LPOAP due to its power-saving mechanism. We did not simulate the power consumption of RAP, GRAP and GRAPO because a) these have worse performance than LPOAP for medium and high loads as indicated by the performance results and b) every mobile station must be constantly "awake" in these protocols as it has no way to know in advance whether it will be or not the intended receiver of the packet transmitted by the polled mobile station. Thus, these protocols cannot support a DOZE state.

We simulated two configurations: Network N1, a 10-mobile station network characterized by low-grade bursty traffic (mean burst length and station buffer size both equal to 10 data packets) and Network N2, a 5-mobile station network characterized by high-grade bursty traffic (mean burst length equal to 1000 data packets and station buffer size of 3 data packets). The bursty traffic is produced as in [1].

In our simulation model, the condition of any wireless link was modeled using a finite state machine with three states [1]. State G, denotes that the wireless link is in a relatively "clean" condition and is characterized by a small BER, given by the parameter G_BER and the corresponding Data Packet Error Rate is given by the parameter G_DPER. State B, denotes that the wireless link is in a condition characterized by increased BER, which is given by the parameter B_BER and the corresponding Data Packet Error Rate is given by the parameter B_DPER. State H, denotes that the pair of communicating stations is out of range of one another. The time spent by a link in states G, B and H are exponentially distributed, but with different average values, given by the parameters TG, TB, TH respectively. The status of a link probabilistically changes between the three states. When a link is in state G (or B) and its status is about to change, it transits either to state H, with probability given by the parameter P_h, or to state B (or G), with probability 1-P_h. When a link spent its time in state H, it transits either to state G or B, with the same probability (0.5).

We set $G\_BER = 10^{-10}$, $B\_BER = 10^{-4}$ and the DATA packet size to 6400 bits. This gives $G\_DPER = 0.64 * 10^{-6}$ and $B\_DPER = 0.47$. We also set TG=50 sec, TB=10 sec, TH=5 sec and P_h =0.1. Simulation measurements have shown that this configuration yields an average data packet error rate around 10%. The learning parameter L of the pursuit Learning Automaton was set to 0.9. We also set L_{RAP}=2 [3] and the number of random addresses in RAP, GRAP and GRAPO was set to $P_{RAP} = 5$ [3]. The size of MAC control packets is 160 bits and the overhead for the orthogonal transmission of the random addresses in RAP, GRAP and GRAPO is set to five times the size of the poll packet, as in [3]. The wireless medium bit rate is set to 1 Mbps. The propagation delay between any two stations is set to 1 μsec. Moreover, as done in other low-power MAC protocols [5], power consumption at the TRM, REC, IDLE and DOZE states are 1.65W, 1.4W, 1.15W and 0.045W respectively.

Figures 2-4 present the simulation results we have obtained and lead to the following conclusions:

1) **LPOAP performance superiority**. LPOAP has superior performance (Figures 2 and 3) to RAP, GRAP and GRAPO, as it is collision-free and has less overhead

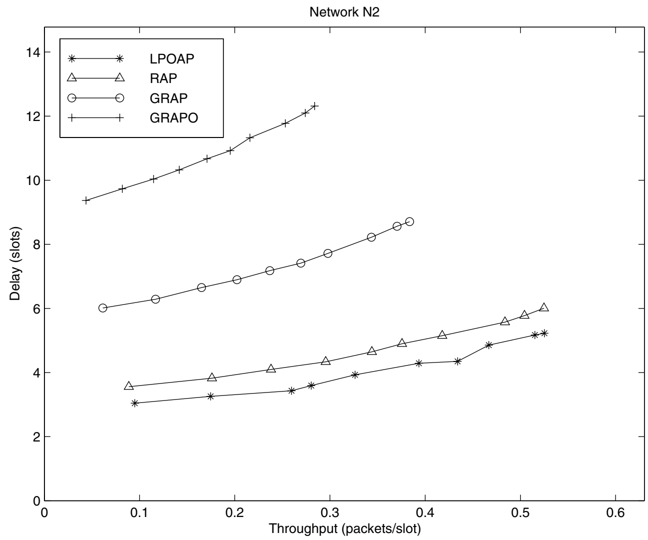

Fig. 3. The delay versus throughput characteristics of LPOAP, RAP, GRAP and GRAPO when applied to network N2.

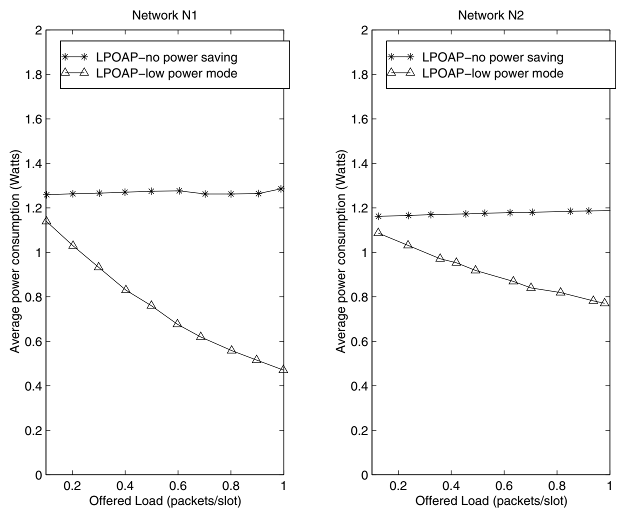

Fig. 4. Average mobile power consumption for LPOAP.

for a DATA packet transmission. LPOAP requires an overhead of three control packets per DATA packet (POLL, BUFF_DATA, ACK). RAP, GRAP and GRAPO can transmit at most five DATA packets per with an overhead of sixteen control packets, for L_{RAP}=1, (READY, CDMA transmission of random addresses which is equal to five times the duration of a control packet, five POLL packets, five ACK packets) yielding an overhead of 3.2 control packets per DATA packet. However, this scenario seldomly occurs due to the increasing number of collisions in RAP when the number of active stations per polling cycle approaches the number of random addresses, P_{RAP}. This happens in high loads and in such cases the increased number of collisions a) reduces the throughput of RAP at high loads and b) at the same time the average delay is also increased a lot [1]. These two remarks explain the curve of RAP at high loads in Figure 2. Finally, under heavy bursty traffic conditions (Figure 3) the number of active stations per polling cycle is significantly less than P_{RAP}, resulting in increased overhead per DATA packet for RAP, GRAP and GRAPO when compared to LPOAP.

2) **Increased power efficiency of LPOAP**. When LPOAP utilizes the BUFF_DATA control packet to inform mobile stations to go to the DOZE power state, its average power consumption is significantly reduced. This can be seen from Figure 4. These power savings increase for an increasing load as in such conditions, the learning mechanism almost always polls mobile stations that have a buffered packet, a fact that leads the other mobile stations to go to the DOZE state. Contrary, in low offered loads, polled stations are most of the time idle and thus the other mobile stations seldomly go to the DOZE state. Moreover, we see more power gains in N1 than N2 because in networks with more mobile stations (N1), each data packet transmission from a polled mobile station results to a transition to the DOZE state of a larger percentage of mobile stations. Finally, the low-power operation does not impact the performance of LPOAP, as the results in Figures 2, 3 were obtained for the low-power mode of LPOAP and are identical to those we obtained for LEAP. This is a positive property as although it is indeed desirable it is not always possible for MAC protocols [6].

IV. Conclusion

This paper proposed the LPOAP MAC protocol for infrastructure WLANs. It operates efficiently under burtsy traffic and can reduce mobile power consumption without performance penalties.

References

[1] P. Nicopolitidis, G. I. Papadimitriou, and A. S. Pomportsis, "Learning-automata-based polling protocols for wireless LANs," *IEEE Trans. Commun.*, vol. 51, no. 3, pp. 453-463, March 2003.
[2] M. A. L. Thathachar and P. S. Sastry, "Estimator algorithms for learning automata," in *Proc. Platinum Jubilee Conference on Systems and Signal Processing*, Bangalore, India, Dec. 1986.
[3] K. C.Chen, "Medium access control of wireless LANs for mobile computing," *IEEE Network*, pp. 50-63, Sept./Oct. 1994.
[4] M.-C. Li and K.-C. Chen, "GRAPO-optimized group randomly addressed polling for wireless data networks," in *Proc. 44th IEEE VTC*, June 1994, pp. 1425-1429.
[5] T. D. Lagkas, G. I. Papadimitriou, P. Nicopolitidis, and A. S. Pomportsis, "A new approach to the design of MAC protocols for wireless LANs: combining QoS guarantee with power saving," *IEEE Commun. Lett.*, vol. 10, no. 7, pp. 537-539, July 2006.
[6] Y.-K. Sun and K.-C. Chen, "Energy-efficient multiple access protocol design," IEEE Commun. Lett., vol. 2, no. 8, pp.334-336, Dec. 1998.